Titre: Title:	SAT : Sécurisation de l'ADS-B grâce à TESLA
Auteur: Author:	Paul Berthier
Date:	2017
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Berthier, P. (2017). SAT : Sécurisation de l'ADS-B grâce à TESLA [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/2540/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/2540/
Directeurs de recherche: Advisors:	Jean-Marc Robert, & Jose Manuel Fernandez
Programme: Program:	Génie informatique

UNIVERSITÉ DE MONTRÉAL

SAT : SÉCURISATION DE L'ADS-B GRÂCE À TESLA

PAUL BERTHIER
DÉPARTEMENT DE GÉNIE INFORMATIQUE ET GÉNIE LOGICIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE INFORMATIQUE)
AVRIL 2017

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

SAT : SÉCURISATION DE L'ADS-B GRÂCE À TESLA

présenté par : BERTHIER Paul
en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées
a été dûment accepté par le jury d'examen constitué de :

M. BELTRAME Giovanni, Ph. D., président
M. FERNANDEZ José M., Ph. D., membre et directeur de recherche
M. ROBERT Jean-Marc, Ph. D., membre et codirecteur de recherche
M. QUINTERO Alejandro, Doctorat, membre

DÉDICACE

La déesse bée d'admiration. Déesse des airs !
Avions-nous prévu ces mots d'espoir ?
Ces curies technocrates en voient chaque jour.
Ah vie ! Ah si on échappait à tes rites sages !
Ce qu'on fit en secret prend son envol . . .

REMERCIEMENTS

Je souhaite remercier toutes les personnes qui m'ont accompagné au cours de ma maîtrise. Tout d'abord José M. Fernandez, mon directeur de recherche. Il m'a accueilli au milieu de mon parcours, et m'a permis de travailler sur un sujet qui me tient à cœur. Jean-Marc Robert, mon codirecteur de recherche, m'a quant à lui beaucoup aidé grâce à ses relectures d'article et de mémoire, et nos discussions autour d'un bon café !

Je dois également souligner l'apport précieux des pilotes François Blais et Hans Obas, qui ont accepté de se déplacer à Polytechnique afin de nous faire part de leurs commentaires et suggestions quant au choix des paramètres optimaux de SAT.

Je tiens bien sûr à remercier toute l'équipe du laboratoire de recherche pour leurs conseils et la bonne humeur qu'ils apportent ! François Labrèche, avec qui j'ai partagé de longues heures de travail. La "râlerie à la française" est un art qui n'a plus aucun secret pour lui et il le pratique désormais avec aisance. Les massages de Nedra Hamouda furent très appréciés après les interminables journées d'écriture. Merci à elle d'avoir supporté mes quelques nombreux moments de mauvaise humeur ! François Menet "L'Illuminé" m'a aidé lors de mes réflexions sur la sécurité de SAT - qui, si j'avais suivi ses conseils, aurait un nom bien moins conventionnel. Nader Ammari m'a également fait me questionner sur de nombreux problèmes potentiels du protocole, et m'a accompagné lors de mes samedis soir de travail. Je remercie également Fanny pour sa joie de vivre et ses formidables sorties de lab, Marwen Jadla pour le partage de ses vidéos et de sa musique, Matthieu Faou qui m'a précédé au sein du laboratoire et m'a donné l'envie d'y rentrer, et Antoine Lemay pour ses délicieux gâteaux qui ont rendu les réunions de lab incontournables ! Je remercie également Militza Jean pour la relecture de mon article, sa bonne humeur, et sa gentillesse.

Je désire enfin remercier Joana Casenave pour la relecture de mon mémoire et pour tous ses encouragements, qui m'ont été très précieux. Je salue également toute ma famille, sans qui je ne me serais pas rendu aussi loin dans mes études. Je pense tout particulièrement à ma sœur Hélène et à mon cousin Vladislav, qui porte fièrement le chandail de Polytechnique que je lui ai offert.

RÉSUMÉ

L'époque où l'unique moyen de connaître la position des avions était l'utilisation de radars est révolue. Désormais, les avions sont équipés de systèmes de positionnement par satellite très précis. Depuis 2010, une nouvelle technologie, l'ADS-B, leur permet alors de transmettre leur position directement aux contrôleurs par ondes radio. Cette information est également diffusée à tous les avions aux alentours.

Cependant, l'ADS-B n'intègre aucun mécanisme d'authentification ni d'intégrité des messages. Cela signifie qu'un attaquant, possédant pour quelques centaines de dollars de matériel électronique et des connaissances basiques sur le fonctionnement du protocole, peut très facilement émettre des messages pour se faire passer pour un avion aux yeux des contrôleurs. Il peut également modifier la trajectoire que ces derniers perçoivent d'un avion réel. De nombreux chercheurs se sont penchés sur cette problématique au cours des dernières années, cependant aucune des méthodes proposées ne prend totalement en compte toutes les contraintes relatives au contexte spécifique de la navigation aérienne.

Parmi les méthodes traditionnelles, la cryptographie à clés symétriques ne convient pas, car elle suppose de partager la clé servant à l'authentification avec tous les récepteurs potentiels. Ces derniers seraient alors capables de se faire passer pour l'avion propriétaire de la clé. La cryptographie à clé asymétrique, quant à elle, utilise des clés et signatures relativement longues. Cela consomme une grande quantité de bande passante, alors que celle-ci est très limitée.

C'est pourquoi nous proposons l'utilisation du protocole TESLA. Celui-ci combine les avantages de faible consommation de bande passante de la cryptographie symétrique, et de non-divulgation de la clé servant à la production des signatures de la cryptographie asymétrique. Cela est possible grâce à la division du temps en intervalles. Une clé servant à authentifier des messages au cours d'un intervalle ne sera dévoilée qu'au cours d'un intervalle futur. Elle ne sera alors plus valide pour authentifier de nouveaux messages, et aura été remplacée par une nouvelle clé.

L'objectif de ce mémoire est donc de déterminer l'applicabilité, la performance, et la viabilité du protocole TESLA en ce qui concerne la sécurisation de l'ADS-B. Pour cela, nous avons tout d'abord introduit nos propres modifications au protocole afin de l'adapter spécifiquement au contexte du contrôle du trafic aérien. Nous avons notamment remplacé le mécanisme de synchronisation temporelle de TESLA - permettant de savoir dans quel intervalle chacun des avions se trouve - par l'utilisation du temps obtenu par satellites. Nous avons également mis en

place une infrastructure à clés publiques basée sur la structure organisationnelle du contrôle du trafic aérien, avec en haut de l'échelle l'Organisation de l'Aviation Civile Internationale. Le protocole TESLA ainsi modifié est nommé SAT.

Puisque le protocole TESLA tire son asymétrie d'un décalage temporel entre le moment où les messages sont diffusés et celui où la clé ayant servi à l'authentification est distribuée, cela introduit un délai équivalent avant que les messages puissent être authentifiés. Plus les clés sont distribuées régulièrement, plus le délai est réduit, mais plus l'utilisation de bande passante est importante. Nous avons donc dû effectuer des simulations afin de déterminer le délai le plus court possible permettant de respecter la spécification de l'ADS-B relative à la probabilité de collision des messages et à la fréquence de mise à jour de la position des avions du point de vue des contrôleurs. Nous avons alors obtenu un délai optimal de cinq secondes.

Enfin, nous avons effectué des tests d'implémentation sur radio logicielle. En effet, notre solution doit pouvoir fonctionner en parallèle de l'ADS-B traditionnel pendant la période de transition. De plus, les transpondeurs actuels doivent pouvoir être compatibles avec notre protocole par une simple mise à jour, car le coût de remplacement de ceux-ci serait trop important. Nos tests se sont révélés concluants, mais d'autres sont nécessaires avant l'obtention de la certification nécessaire à une utilisation en vol des transpondeurs SAT, et de l'adoption du protocole par l'OACI comme standard international.

ABSTRACT

Radars used to be the only way to get the position of planes from the ground. But planes are now equipped with a satellite navigation system which is very accurate. Since 2010, a new technology named ADS-B has allowed them to directly send their position to controllers thanks to a radio signal. Surrounding planes are also able to get the information.

However, no authentication nor message integrity scheme is implemented in ADS-B. This means an attacker - with as little as a few hundred dollars of electronics and basic knowledge of the protocol - can easily send messages in order to impersonate a plane. He can also modify the trajectory of a real plane as seen by the controllers. Several researches have tried to tackle this problem for the past years, but none of them has really taken into account all of the constraints relative to the specific context of air traffic control.

Among common methods, symmetric cryptography can't be used because it makes necessary to share the authentication key with all the potential receivers. Each of them could therefore impersonate the owner of the key. As for asymmetric cryptography, it uses long keys and signatures. This leads to a significant utilization of the scarce bandwidth.

We therefore suggest the use of TESLA. It is a protocol which combines both the advantages of symmetric cryptography - low bandwidth usage - and asymmetric cryptography - not sharing the key used for producing message signatures. This is made possible by dividing the time in intervals. A key used to authenticate messages for the duration of an interval will only be unveiled during a later interval. At this time, it won't be valid anymore for signing messages, but will have been replaced by a new key.

This thesis has for objective to evaluate the relevance, the performance and the viability of TESLA for securing ADS-B. We first modified the protocol in order to make it suitable for air traffic control. We replaced the synchronization mechanism of TESLA - which makes possible to know in which interval a plane is - by the use of satellite time. We also set up a public key infrastructure based on the air traffic control hierarchy, with the ICAO at the top. The modified TESLA protocol is named SAT.

The source of TESLA asymmetry comes from the time difference between the moment when a message is broadcast and the moment when the key used for authentication is unveiled. This causes a delay before messages can be authenticated. The more frequently the keys are broadcast, the shorter the delay, but the more the bandwidth is used. We therefore carried out some simulations in order to determine the shortest delay possible so that the specifica-

tion of ADS-B regarding the messages collision probability and position update frequency is respected. We found an optimal delay of five seconds.

Last, we performed some implementation tests on software defined radios. Our protocol has to work along with traditional ADS-B during the transition period. Moreover, current transponders have to be compatible with SAT thanks to a software update, because the cost of replacing them would be too significant. Our tests were conclusive, but further trials are necessary in order to pass the certification process. This will allow SAT transponders to be used in-flight, and the final objective is to make our solution adopted as an international standard by the ICAO.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

ACARS Aircraft Communication Addressing and Reporting System

ADS-B Automatic Dependant Surveillance – Broadcast

AES Advanced Encryption Standard

ANSSI Agence Nationale de la Sécurité des Systèmes d'Information

ASDI Aircraft Situational Display to Industry

ATC Air Traffic Control

ATCRBS Air Traffic Control Radar Beacon System

CA Certificate Authority

CAA Civil Aviation Authority

CDMA Code Division Multiple Access

CPDLC Controller-Pilot Data Link Communication

CPR Compact Position Reporting

CRC Contrôle de Redondance Cyclique

CRL Certificate Revocation List

DGAC Direction Générale de l'Aviation Civile

DME Distance Measuring Equipment

DSA Digital Signature Algorithm

DoS Denial of Service

EASA European Aviation Safety Agency

ECDSA Elliptic Curve Digital Signature Algorithm

ES Extended Squitter

ESCAT Emergency Security Control of Air Traffic

FAA Federal Aviation Administration

FIS-B Flight Information System – Broadcast

FPE Format-Preserving Encryption

GDOP Geometric Dilution Of Precision

GLONASS GLObalnaya NAvigatsionnaya Sutnikovaya Sistema

GNSS Global Navigation Satellite System

GPS Global Positioning System

HMAC keyed-Hash Message Authentication Code

IBS Identity-Based Signature

ICAO International Civil Aviation Organization

ICP Infrastructure à clés publiques

IFF Identification Friend or Foe

IFR Instrument Flight Rules

LDACS L-band Digital Aeronautical Communications System

LEO Low Earth Orbit

MAC Message Authentication Code

MLAT MultiLATeration

NAA National Aviation Authority

NATS National Air Traffic Services

NIST National Institute of Standards and Technology

OACI Organisation de l'Aviation Civile Internationale

ONU Organisation des Nations Unies

OTAN Organisation du Traité de l'Atlantique Nord

PKD Public Key Directory

PKG Private Key Generator

PKG Private Key Generator

PKI Public Key Infrastructure

PPM Pulse Position Modulation

PRF Pseudo Random Function

PSR Primary Surveillance Radar

RADAR RAdio Detection And Ranging

RDF Resource Description Framework

RTCA Radio Technical Commission for Aeronautics

SAT Security in the Air using TESLA

SCATANA Security Control of Air Traffic and Air Navigation Aids

SDR Software Defined Radio

SPARQL SPARQL Protocol and RDF Query Language

SSR Secondary Surveillance Radar

TACAN TACtical Air Navigation

TCAS Traffic Collision Avoidance System

TDMA Time Division Multiple Access

TDOA Time Difference Of Arrival

TESLA Timed Efficient Stream Loss-tolerant Authentication

TIS-B Traffic Information Service – Broadcast

TOA Time Of Arrival

UAT Universal Access Transponder

USRP Universal Software Radio Peripheral

UTC Coordinated Universal Time

VDOP Vertical Dilution Of Precision

VFR Visual Flight Rules

WAAS Wide Area Augmentation System

LISTE DES ANNEXES

CHAPITRE 1 INTRODUCTION

1.1 Historique du contrôle du trafic aérien

Les faits historiques décrits ci-après sont extraits du livre de Nolan (2010).

L'histoire du contrôle du trafic aérien, ou Air Traffic Control (ATC), commence au début des années 1930. À cette époque, un homme se plaçait sur la piste et agitait des drapeaux de couleur afin de donner des indications aux pilotes.

À la même époque, une première génération de radars fit son apparition, sous le nom de Primary Surveillance Radars (PSRs). Cependant, ceux-ci furent tout d'abord destinés à un usage militaire. Une antenne rotative émet une onde qui, lorsqu'elle rencontre un avion ou un objet quelconque dans le ciel, se réfléchit. L'onde réfléchie est alors reçue par le radar. Il est possible de connaître la direction de l'objet ayant causé la réflexion en regardant l'orientation de l'antenne. On peut également déduire sa distance à partir du temps qu'a mis l'onde pour faire un aller-retour entre le radar et l'objet à la vitesse de la lumière - l'onde émise étant une onde électromagnétique.

Les PSR avaient cependant une grande limitation : ils n'étaient pas capables de distinguer les avions « ennemis » des avions « amis ». C'est pourquoi la Seconde Guerre mondiale entraîna l'apparition d'une nouvelle technologie : l'Identification Friend or Foe (IFF). On passe alors d'un système complètement passif du point de vue de l'avion à un système actif. En effet, celui-ci embarque désormais un transpondeur, c'est-à-dire un appareil capable de répondre à des interrogations envoyées par un transmetteur au sol. Ces interrogations se présentent sous la forme de pulsations séparées par des intervalles de temps prédéterminés. Le transpondeur est configuré pour répondre à certains types d'interrogations par un code spécifique. Les opérateurs au sol étaient dés lors capables de distinguer un avion ami si celui-ci répondait par le code adéquat à leurs interrogations. Dans le cas contraire, l'avion était considéré comme étant ennemi. Les antennes émettrices des interrogations de l'IFF furent bientôt intégrées directement au sein des radars. L'ensemble fut alors nommé radar secondaire, ou Secondary Surveillance Radar (SSR).

Peu après la fin de la Seconde Guerre mondiale, des expérimentations commencèrent aux États-Unis afin d'adapter le radar aux besoins de l'aviation civile. Cependant, des coupures budgétaires sur les moyens accordés au développement du contrôle du trafic aérien mirent en pause le programme. Il faut attendre le 30 juin 1956, et la collision de deux avions de ligne au-dessus du Grand Canyon, pour que le Congrès américain débloque des fonds pour moder-

niser le système d'ATC. La Federal Aviation Administration (FAA) fut ainsi créée afin de le superviser de manière indépendante. Les SSR commencèrent alors à être largement déployés pour un usage civil sous le nom d'Air Traffic Control Radar Beacon System (ATCRBS), et un certain nombre d'améliorations leur furent apportées afin de les adapter au monde civil. L'IFF fut notamment peu à peu étendue sous la forme de plusieurs modes. En effet, il ne s'agissait désormais plus de distinguer un avion ami ou ennemi, mais d'identifier chaque avion (mode A) et d'obtenir son altitude (mode C) ; ceci afin de pouvoir le suivre pendant toute la durée de son vol et de s'assurer qu'il est toujours à une distance sécuritaire des autres appareils. Enfin, dans les années 1990, un nouveau mode fut introduit : le mode S. Celui-ci permet à un avion de transmettre encore plus d'informations à la tour de contrôle, comme sa position et sa vitesse.

Cependant, les radars ont un certain nombre d'inconvénients. Ils coûtent cher, et ont besoin d'une maintenance régulière. De plus, ils ont une portée limitée, et leur performance est réduite en cas de mauvaises conditions météo (la pluie perturbe la diffusion des ondes radio). En conséquence, la FAA commença à envisager au début des années 2000 un plan de grande envergure visant à l'évolution de la surveillance par radars. Les avions étant désormais équipés d'un système de navigation par satellite fournissant de manière très précise position, altitude et vitesse, seule la partie IFF des SSR permettant la transmission de ces informations à travers le mode S reste utile, le rôle du PSR devenant obsolète. L'idée est alors de remplacer les SSR par un système conjoint de satellites et d'antennes au sol permettant la réception du Mode S (Maxson, 2011). Ces travaux aboutirent en 2010 à la définition d'un nouveau protocole : l'Automatic Dependant Surveillance – Broadcast (ADS-B). Contrairement à ce qui se faisait avec les SSR, les transpondeurs utilisant ce nouveau protocole transmettent de manière périodique leurs informations par Mode S, sans qu'aucune requête ne leur soit envoyée. Cela permet non seulement aux contrôleurs, mais désormais aussi aux autres avions à proximité de connaître en temps réel la position et la vitesse de l'avion utilisant l'ADS-B.

NAV CANADA, la société fournissant les services d'ATC au Canada, fut parmi les premières au monde à déployer un réseau d'antennes ADS-B à travers la baie d'Hudson (Wright, 2009). En effet, cette zone désertique du nord du Canada, où il était très difficile d'installer des radars, est pourtant survolée par plus de 35 000 avions par an (Taylor, 2015). Il était donc important d'y fournir un service de contrôle du trafic aérien, ce qui était impossible avant l'arrivée de l'ADS-B. Il en est de même pour la route transatlantique, très empruntée, mais ne pouvant être contrôlée par radar, car située au-dessus de l'océan. La solution est ici de mettre en place une constellation de 66 satellites en orbite basse (Low Earth Orbit, LEO), nommée *Aireon*, visant à recevoir les messages ADS-B émis par les avions. Ce projet est une initiative conjointe des agences de contrôle du trafic aérien du Canada (NAV CANADA), de

l'Irlande (IAA), de l'Italie (ENAV), et du Danemark (Naviair) en partenariat avec la société *Iridium Communications* (Aireon, 2016; NAV CANADA, 2016).

Aux États-Unis, la FAA s'est lancée dans un programme de grande ampleur de modernisation de son système d'ATC, nommé *NextGen*. Son objectif principal est de permettre l'augmentation du trafic aérien en réduisant l'espacement entre les avions, tout en améliorant la sécurité des vols et en contrôlant les coûts. Une des composantes principales de NextGen est la généralisation de l'ADS-B, dont la précision des données de position des avions est meilleure que celle des radars. La majorité de l'espace aérien des États-Unis est déjà couvert par des antennes ADS-B, et la couverture sera complétée par satellite au fur et à mesure du déploiement de la constellation *Aireon*. En ce qui concerne l'équipement des avions, celui-ci s'accélère, car il sera obligatoire à partir de 2020 d'émettre de l'ADS-B afin d'être autorisé à survoler les zones où il était auparavant nécessaire d'être équipé d'un transpondeur (U.S. Government Publishing Office, 2017) - soit une large partie de l'espace aérien américain.

1.2 Problématique

La grande évolution de l'ADS-B par rapport aux PSRs consiste en ce que les informations de position des avions ne proviennent plus d'une mesure physique directe depuis le sol (le temps que met une onde pour se réfléchir sur un objet), mais d'une mesure effectuée au sein même de l'avion. Celle-ci est ensuite transmise au sol de manière encodée par l'intermédiaire d'un canal de communication sans fil. Par rapport aux SSRs, l'ADS-B ne fait que conserver la partie IFF. L'envoi de requêtes est également rendu inutile, car les avions diffusent désormais leurs informations de manière périodique et spontanée. Toutes les données de vol nécessaires aux contrôleurs sont obtenues grâce à l'avionique et aux systèmes de navigation de l'avion. Cela comprend les informations en provenance du système de navigation par satellite, dont la dénomination générique est le Global Navigation Satellite System (GNSS).

Le premier point indispensable au bon fonctionnement de l'ADS-B est la coopération du pilote. En effet, si celui-ci décide de fausser les informations émises par son transpondeur ou de couper ce dernier, il n'est alors plus possible de le suivre sur les écrans des contrôleurs sans avoir recours aux PSRs. Cependant, en opérations normales, un pilote n'a aucun intérêt à interférer avec son transpondeur. Il coopère avec les contrôleurs aériens afin de faire parvenir son avion à destination en toute sécurité. Les seuls cas connus de coupure du transpondeur furent en cas d'attaques terroristes, notamment le 11 septembre 2001. Le pirate de l'air veut alors disparaître des écrans afin de pouvoir effectuer son attaque sans être intercepté. Cependant, des procédures existent pour ces cas spécifiques. Le pilote peut déclencher une alerte par l'intermédiaire de codes d'urgence qu'il rentrera sur son transpondeur ou qu'il

transmettra par radio. Nous n'examinerons donc pas ce cas de figure d'une attaque interne à l'avion.

Un second type d'attaque pourrait être effectué sur les instruments de navigation et plus spécifiquement sur le système de GNSS, ceci afin de fausser la position calculée par l'avion. Ce type d'attaque est hors de la portée de ce mémoire, mais de nombreux travaux ont été consacrés à ce problème et le lecteur intéressé pourra notamment se référer aux travaux de Papadimitratos et Jovanovic (2008). Nous ferons désormais l'hypothèse que chaque avion a son transpondeur activé en permanence et que celui-ci transmet des informations véridiques (les données en provenance des instruments de navigation ne sont pas altérées).

Cependant, même s'il n'y a pas de problème quant à la fiabilité des données de navigation à la source, reste le problème de la transmission de ces informations. Celle-ci se fait par ondes radio sans aucune protection cryptographique, et peut donc être altérée de différentes manières. Premièrement, n'importe qui possédant un minimum d'équipement électronique a les capacités d'émettre de l'ADS-B, et de se faire passer pour un avion même en restant au sol. On peut également, en émettant un signal sur la bonne fréquence et avec suffisamment de puissance, bloquer toute transmission. On appelle cela le brouillage radio ou *jamming*. Il est aussi possible d'intercepter les messages échangés et de connaître en temps réel la position de tous les avions. On peut enfin interférer avec les messages qu'envoie un avion en particulier afin d'altérer sa trajectoire aux yeux des contrôleurs.

Ces différents problèmes de sécurité ont déjà été évoqués à maintes reprises (Haines, 2012; Strohmeier et al., 2014, 2015a; McCallie et al., 2011; Wesson et al., 2014). Cependant, les différentes instances gouvernementales et internationales chargées de la sécurité du trafic aérien ne semblent que modérément inquiètes. L'Organisation de l'Aviation Civile Internationale (OACI) explique que la plupart des articles publiés dans la presse traitant de la sécurité de l'ADS-B ne considèrent pas tous les paramètres rentrant en compte pour l'utilisation des données de surveillance par les contrôleurs aériens (APANPIRG/26, 2015). Leur argument est que les données en provenance de l'ADS-B ne sont pas rentrées telles quelles dans le système de contrôle du trafic aérien et affichées aux contrôleurs. Elles sont auparavant croisées avec les autres informations disponibles, notamment celles en provenance des PSR et SSR, afin de s'assurer de leur cohérence. Cela implique néanmoins de conserver un nombre important de radars en service alors que les plans initiaux de nombreuses agences, dont la FAA, étaient d'économiser en diminuant leur nombre (Wesson et al., 2014).

Cependant, les vérifications croisées des données de positionnement ne résolvent pas tous les problèmes. Tout d'abord, cela ne permet pas d'assurer la sécurité là où la surveillance est rendue nouvellement possible par l'arrivée de l'ADS-B, c'est-à-dire dans des régions où

l'installation de radars est impossible. C'est le cas des régions désertiques ou océaniques, où seule la présence d'antennes ADS-B et de satellites permet la réception de la position des avions en temps réel. De plus, l'un des objectifs du plan *NextGen* de modernisation du système d'ATC est d'augmenter les performances de celui-ci. Comme les données reçues par ADS-B sont plus précises que celles des radars, une diminution de l'espacement entre les avions est rendue possible. Cependant, ceci n'est valable que si l'on peut s'assurer de la validité des données ADS-B.

1.3 Comment sécuriser l'ADS-B ?

La *sécurisation* d'un échange d'informations peut avoir différentes significations. On peut vouloir parler de *confidentialité*, c'est-à-dire s'assurer que les données échangées ne soient intelligibles qu'aux intervenants légitimes - généralement l'émetteur et le récepteur. On peut également rechercher *l'authentification*, ce qui signifie avoir la certitude que l'entité avec qui l'on échange est bien celle que l'on pense. On peut encore vouloir s'assurer de *l'intégrité* des informations échangées, c'est-à-dire être certain que les données que l'on reçoit sont bien celles qui ont été envoyées initialement, sans altération.

Pour parvenir à chacun de ces objectifs, on peut penser agir sur le médium physique. Si l'on est capable d'en limiter l'accès aux seules personnes autorisées, aucun problème ne se pose. Cependant, cela n'est pas envisageable dans le cas de transmissions sans fil. On peut également penser à la technique du saut de fréquence, qui consiste à utiliser alternativement plusieurs fréquences dans une bande donnée et selon une séquence pseudo-aléatoire connue uniquement de l'émetteur et du récepteur. Cependant, cela demanderait de complètement repenser le système, qui ne fonctionne actuellement que sur deux fréquences : 1090 MHz et 978 MHz. Nous n'aborderons donc pas plus en détail cette possibilité par la suite.

L'autre solution consiste à agir sur les données. On peut par exemple avoir recours à différentes techniques cryptographiques, comme la cryptographie symétrique, la cryptographie asymétrique ou encore les fonctions de hachage. Chacune d'entre elles présente ses propres caractéristiques et permet de répondre à un sous-ensemble des objectifs énoncés précédemment. Ces différentes alternatives seront discutées dans le chapitre 2 de ce mémoire.

Après avoir étudié les différentes possibilités offertes par les méthodes cryptographiques traditionnelles et les travaux antérieurs liés à la sécurisation de l'ADS-B, nous sommes parvenus à la conclusion qu'aucun d'entre eux ne possédait les caractéristiques nécessaires à une utilisation dans le contexte du contrôle du trafic aérien. Il faut bien entendu tout d'abord s'intégrer dans l'écosystème aéronautique. Cela signifie comprendre les procédures en place,

et s'y conformer. Il ne faut également pas oublier que l'avionique coûte très cher, car tout équipement à bord d'un avion doit être certifié, et donc sa fréquence de renouvellement est très faible. De plus, l'adoption d'un nouveau protocole prend également beaucoup de temps. En conséquence, la solution proposée devrait pouvoir être rétrocompatible, et suffisamment évolutive pour rester en place de nombreuses années. De plus, elle devra être compatible avec le fonctionnement en mode *broadcast* de l'ADS-B, c'est-à-dire que n'importe quel avion doit pouvoir être capable d'identifier les autres avions et leur position. Ceci n'est par exemple pas directement faisable avec de la cryptographie symétrique classique. Enfin, les bandes de fréquence utilisées par l'ADS-B étant déjà très chargées, l'évolution du protocole que nous proposerons ne devra pas utiliser une grande quantité de bande passante. En particulier, la cryptographie asymétrique classique ne paraît pas appropriée du fait de la longueur de ses clés qu'il est nécessaire d'échanger.

En conséquence, nous nous sommes intéressés au protocole Timed Efficient Stream Loss-tolerant Authentication (TESLA) de Perrig et al. (2002a), qui semble répondre à tous ces critères. L'idée d'utiliser ce protocole avait déjà été proposée par Strohmeier et al. (2015a), sans toutefois en détailler une possible implémentation. Cette solution, initialement développée pour les réseaux de capteurs, cumule les avantages des systèmes cryptographiques symétriques et asymétriques. Elle permet d'obtenir une authentification des différents acteurs en mode *broadcast* (ce que permet uniquement la cryptographie asymétrique) et en utilisant peu de bande passante (avantage de la cryptographie symétrique, qui utilise des clés moins longues que la cryptographie asymétrique pour un niveau de sécurité équivalent). Cette version sécurisée de l'ADS-B utilisant TESLA sera nommée Security in the Air using TESLA (SAT).

1.4 Objectif et questions de recherche

L'objectif de ce mémoire sera de déterminer l'applicabilité, la performance, et la viabilité du protocole TESLA en ce qui concerne la sécurisation de l'ADS-B ; en concordance avec la problématique formulée précédemment. Afin d'y parvenir, différentes questions de recherche seront soulevées :

Q.1 En quoi les solutions antérieures proposées pour sécuriser l'ADS-B ne sont-elles pas satisfaisantes ?

Q.2 Quelles sont les spécificités du protocole TESLA qui en font une solution intéressante pour la sécurisation de l'ADS-B ?

Q.3 Quelles sont les modifications à apporter à TESLA pour qu'il puisse fonctionner avec l'ADS-B, dans le contexte très spécifique du contrôle du trafic aérien ? Quels sont les points techniques d'implémentation à considérer ?

Q.4 Jusqu'à quel point une augmentation de la quantité de données nécessaires à notre version sécurisée de TESLA est-elle envisageable ? Quel en serait l'impact sur le taux de succès de réception des messages, et donc sur la fiabilité du protocole - critique dans le cadre du transport aérien ?

Q.5 Quelle serait la facilité d'adoption du protocole SAT ? En particulier, quel serait le niveau de compatibilité avec les transpondeurs actuels ?

1.5 Plan du mémoire

Dans un premier temps, le chapitre 2 présentera les concepts importants liés à la sécurité de l'ADS-B. Les différents acteurs de l'ATC et le fonctionnement du protocole seront tout d'abord présentés, puis les problèmes de sécurité et les menaces associées seront exposés. Nous illustrerons ces menaces par des exemples concrets d'attaques potentielles, puis détaillerons les objectifs de sécurité que devraient rechercher les différentes instances gouvernementales. Nous présenterons enfin les travaux antérieurs, et expliquerons en quoi ils ne répondent qu'en partie à ces objectifs ou bien ne prennent pas totalement en compte les contraintes techniques de l'ADS-B.

Dans un second temps, nous présenterons au chapitre 3 le protocole TESLA. Nous détaillerons la solution originale telle que décrite par Perrig et al. (2001, 2002a) et son application aux réseaux de capteurs (Perrig et al., 2002b).

Nous présenterons ensuite au chapitre 4 les adaptations que nous proposons pour le protocole TESLA afin qu'il soit utilisable pour sécuriser l'ADS-B. Nous détaillerons en particulier les

différentes méthodes de certification servant au processus d'authentification des messages envoyés.

Par la suite, le chapitre 5 sera consacré à la détermination des paramètres optimaux de TESLA dans le contexte de l'ADS-B. L'objectif principal sera de réduire au maximum l'utilisation de la bande passante, tout en conservant un niveau de service satisfaisant. On devra notamment se conformer aux exigences de la Radio Technical Commission for Aeronautics (RTCA), et aux recommandations d'un panel de pilotes que nous avons consulté.

Le chapitre 6 décrira notre implémentation de SAT à l'aide de radios logicielles, ou Software Defined Radios (SDRs). Nous montrerons la manière dont il est possible de déployer notre solution sur l'avionique actuelle par une simple mise à jour logicielle.

Le chapitre 7 synthétisera enfin l'ensemble des résultats présentés au long de ce mémoire afin de répondre aux différentes questions de recherche que nous avons énoncées. Nous y présenterons également les perspectives de recherche future afin d'améliorer encore SAT et qu'il puisse remplacer l'ADS-B comme nouveau standard de l'ATC.

CHAPITRE 2 SÉCURITÉ DE L'ADS-B

La sécurité de l'ADS-B est un sujet en vogue. Le domaine de l'aéronautique passionne, et en particulier les accidents d'avion sont largement relayés par la presse. Le fait qu'un pirate puisse interférer avec le fonctionnement du contrôle du trafic aérien et qu'il ait la possibilité de causer de tels accidents suscite donc l'intérêt des chercheurs. De nombreux articles ont donc été publiés depuis les premières présentations sur ce sujet aux conférences Black Hat et DEF CON en 2012 (Costin et Francillon, 2012; Haines, 2012). Cependant, un trop grand nombre d'entre eux se contentent de transposer des méthodes existantes en provenance d'autres domaines (notamment l'Internet) sans réellement s'intéresser à l'applicabilité au secteur de l'aéronautique et à toutes ses spécificités. C'est pourquoi nous trouvons important de reprendre et développer dans une première partie de ce chapitre les différents concepts relatifs au contrôle du trafic aérien, déjà évoqués en introduction. Ceci nous permettra dans un second temps de mieux concevoir les différents objectifs de sécurité pouvant s'appliquer à l'ADS-B. Nous les illustrerons par des exemples concrets d'attaques potentielles, puis expliquerons lesquels d'entre eux seront abordés dans ce mémoire. Nous présenterons enfin les différents travaux antérieurs à notre recherche, en nous attachant à expliquer les points qui les rendent inapplicables étant donné les spécificités du contrôle du trafic aérien. Cela nous conduira au cours des chapitres suivants à attacher une attention toute particulière à leur trouver des solutions.

2.1 Acteurs du contrôle du trafic aérien

Les acteurs intervenant dans le contrôle du trafic aérien sont nombreux, et il est important de bien distinguer leurs différents rôles Nolan (2010). Au niveau international, l'OACI - aussi connue sous son nom anglais d'ICAO - a été créée à la fin de la Seconde Guerre mondiale avec comme objectif d'établir des normes visant à la standardisation des règles dans le domaine du transport aérien. Elle est rattachée à l'Organisation des Nations Unies (ONU). Au niveau européen, EUROCONTROL est l'organisation qui a pour mission d'unifier les règles et la gestion de la navigation aérienne entre les différents pays de l'union. Elle travaille sous le contrôle de l'European Aviation Safety Agency (EASA), créée en 2003 pour établir la réglementation en matière de sécurité de l'aviation civile européenne.

Au niveau national, chaque pays possède ensuite sa propre agence chargée des règles en matière de transport aérien et de gestion de l'ATC, qualifiée de National Aviation Authority (NAA). Ainsi, on trouve la FAA aux États-Unis ou encore la Direction Générale de l'Aviation

Civile (DGAC) en France. D'autres pays ont choisi de rendre indépendante et de privatiser la charge du contrôle du trafic aérien. C'est le cas du Canada, où l'on trouve *Transports Canada* qui est responsable des réglementations concernant le transport aérien, mais également NAV CANADA qui s'occupe de manière privée de la gestion de l'ATC. C'est également le cas au Royaume-Uni, où l'on trouve la Civil Aviation Authority (CAA), l'organisme public chargé des réglementations, et la National Air Traffic Services (NATS), le prestataire privé de l'ATC.

En partenariat avec ces différentes autorités, la RTCA, une organisation américaine à but non lucratif, s'occupe de définir les normes techniques auxquelles doivent se conformer les différents équipements utilisés pour le contrôle du trafic aérien. C'est notamment elle qui a défini les caractéristiques du protocole ADS-B (RTCA, 2002).

Pour la suite de ce mémoire, nous illustrerons parfois nos travaux avec des exemples applicables au domaine aérien américain, qui est sous contrôle de la FAA. En effet, c'est l'une des zones au monde qui concentre le plus de vols aussi bien internationaux que domestiques. De plus, avec son plan *NextGen* et l'obligation d'ici 2020 d'être équipé d'ADS-B pour voler dans la majeure partie de l'espace aérien américain, c'est l'un des pays les plus en avance dans le domaine. Toutefois, tous les résultats et procédures que nous présenterons sont applicables pour l'ensemble des 191 états membres de l'OACI.

2.2 Caractéristiques de l'ADS-B

L'ADS-B est une évolution directe du Mode S de réponse aux interrogations des radars secondaires. Il en partage ainsi les différentes caractéristiques techniques au niveau de la transmission du signal, tel que défini par sa spécification (RTCA Free flight Select Committee et al., 2001). Il utilise une modulation de type Pulse Position Modulation (PPM), comme le montre la capture de la transmission d'un paquet à la figure 2.1. Celle-ci comporte une séquence de synchronisation de 8 μs, appelée préambule, suivie de 112 bits de données. Le débit de transmission est de 1 Mbit/s, ce qui signifie qu'un bit est transmis toutes les microsecondes. Le principe de la PPM, illustré à la figure 2.2, est que chaque période de transmission d'un bit est divisée en deux. Si le signal est d'abord à l'état haut puis à l'état bas sur la période, alors un "1" a été transmis. Dans le cas contraire, c'est un "0".

En ce qui concerne la bande de fréquence, c'est celle de 1090 MHz du Mode S qui est utilisée. Elle est nommée Extended Squitter (ES), par opposition au *squawk* que désigne le "S" du Mode S. Ce sont tous deux des termes du jargon aéronautique. Le terme anglais *squawk* est un cri d'oiseau, et représente la réponse d'un avion aux interrogations du SSR. Au contraire, *squit* est le terme d'argot britannique pour la diarrhée, et illustre bien le fait qu'un avion diffuse

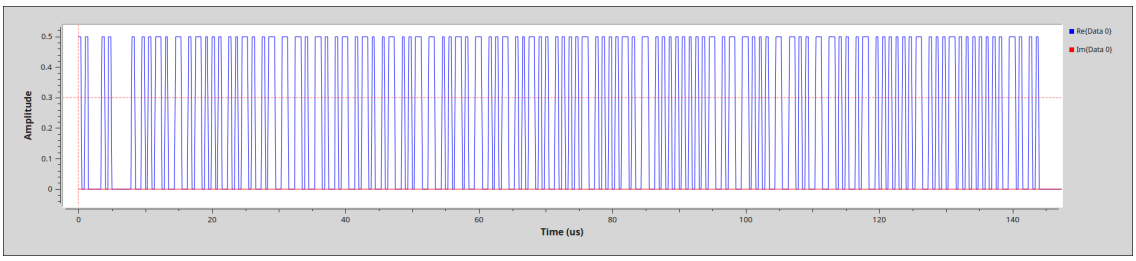


Figure 2.1 Capture de la transmission d'un paquet ADS-B : amplitude du signal reçu en fonction du temps. On reconnait une modulation PPM.

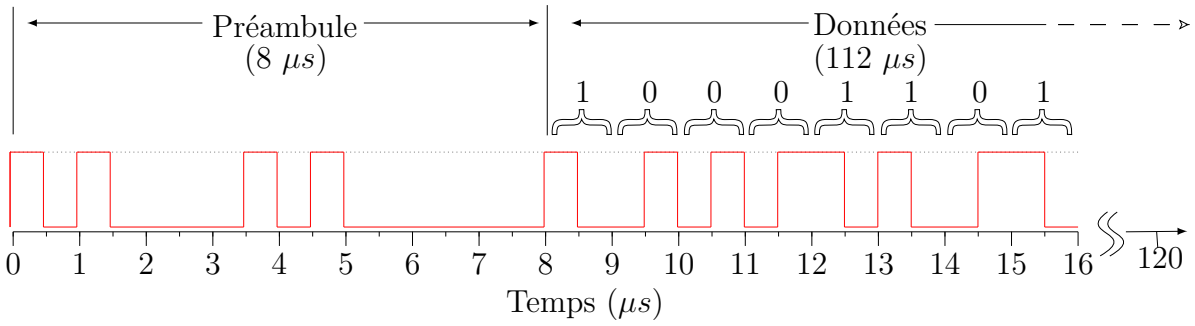


Figure 2.2 Modulation PPM pour un message ADS-B. Le préambule dure 8 μs, puis on retrouve 112 bits de données émis à 1 Mbit/s (soit 112 μs). Chaque période de transmission d'un bit est divisée en deux. Si le signal est d'abord à l'état haut puis à l'état bas sur la période, alors un "1" a été transmis. Dans le cas contraire, c'est un "0".

en continu et sans requête ses messages (les pilotes ont de l'humour). On parle d'*Extended* squitter, car l'ADS-B est capable d'émettre bien plus d'informations que le pouvait le Mode S.

Aux États-Unis, une seconde bande de fréquence, celle de 978 MHz, a également été réservée pour l'usage de l'ADS-B. L'objectif est de désengorger la bande de 1090 MHz, déjà très utilisée (cela sera détaillé dans le chapitre 5 de ce mémoire). Nommée UAT pour Universal Access Transponder, elle a l'avantage de profiter de la bande passante supplémentaire disponible pour implémenter le service Flight Information System – Broadcast (FIS-B) qui diffuse différentes informations utiles aux pilotes, comme la météo locale. Cependant, son usage est restreint aux vols en dessous de 18 000 pieds, et les États-Unis sont les seuls à l'utiliser. Son usage est donc réservé aux vols domestiques à basse altitude, ce qui exclut les avions de ligne (ces derniers volent en général à des altitudes proches de 30 000 pieds, car en altitude l'air est moins dense, la résistance de l'air diminue, et la consommation de carburant en est moindre).

Il existe deux types d'utilisation de l'ADS-B. L'ADS-B *in* consiste à émettre de façon périodique en *broadcast* ses informations de vol. Nous avons vu jusqu'à présent que ces messages

pouvaient être récupérés par les contrôleurs par l'intermédiaire d'antennes au sol ou de satellites. Cependant, les autres avions à proximité peuvent également directement recevoir ces informations s'ils sont équipés de l'ADS-B *in*, qui est une antenne intégrée dans l'avion capable de recevoir les paquets ADS-B. Cependant, il existe deux types de fréquence pour l'ADS-B, et contrairement aux contrôleurs, les avions ne sont généralement équipés que pour en recevoir une seule. En conséquence la technologie Traffic Information Service – Broadcast (TIS-B) a été développée. Le principe est que les contrôleurs réémettent sur chacune des deux fréquences toutes les informations de position des avions qu'ils ont eux-mêmes obtenues, que ce soit sur la bande de 978 MHz, de 1090 MHz, ou bien à partir des radars pour les avions non équipés d'ADS-B *out*. Ainsi, les avions équipés de l'ADS-B *in* peuvent visualiser tout le trafic aérien aux alentours, et mieux anticiper des risques de collision potentiels avant même de prendre contact par radio avec le contrôleur. Les paquets ADS-B peuvent généralement être reçus dans un rayon de 500 km.

Les avions équipés d'ADS-B *out* transmettent donc différentes informations périodiquement. Mais quelles sont ces informations exactement, et à quelle fréquence sont-elles émises ? L'OACI en définit un grand nombre (OACI, 2008). Cependant, seuls certaines d'entre elles sont généralement utilisées actuellement, comme suggéré par la RTCA (RTCA Free flight Select Committee et al., 2001) et comme nous l'avons vérifié en enregistrant un échantillon de messages ADS-B en haut du Mont Royal, à Montréal (ceci sera détaillé au chapitre 6) :

— L'indicatif du vol (p. ex. ACA870), dans le cas d'un avion de ligne. Il est émis généralement toutes les 5 secondes, ou toutes les 2.5 secondes pour certains appareils.

— La position de l'avion, émis deux fois par seconde.

— La vitesse de l'avion, émise deux fois par seconde.

Il est toutefois important de souligner qu'un écart aléatoire compris dans un intervalle de plus ou moins 100 ms est introduit pour l'envoi de chaque message. En effet, il n'y a pas de technique complexe de contrôle d'accès au support de transmission, telle que le Code Division Multiple Access (CDMA) ou le Time Division Multiple Access (TDMA). Il faut donc s'assurer que tous les avions ne transmettent pas en même temps, ce que permet l'ajout de ce temps aléatoire d'un ordre de grandeur nettement supérieur à celui du temps d'émission d'un paquet (120 μs).

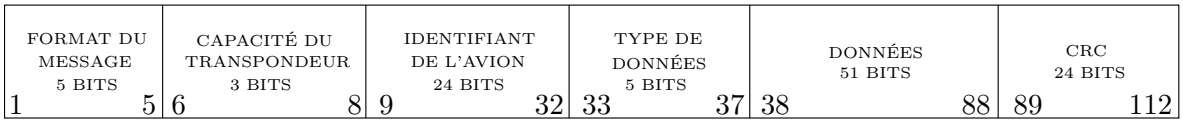


Figure 2.3 Contenu d'un message ADS-B. Il est codé sur 112 bits, précédés par 8 μs de préambule.

Enfin, la structure d'un message ADS-B est présentée à la figure 2.3. Elle est précédée des 8 μs du préambule, et est constituée de :

— Le format du message. Dans notre cas, on envoie le code "17" encodé sur 5 bits, indiquant que l'on transmet un Extended Squitter, soit de l'ADS-B. Il existe d'autres codes pour indiquer par exemple l'envoi d'un message TIS-B, pour répondre aux différentes interrogations des SSR ou encore pour un usage militaire.

— 3 bits indiquant les capacités du transpondeur. Comme le protocole ADS-B a évolué au fil du temps, cela est utile pour savoir par exemple quelle version est utilisée.

— Un identifiant unique de l'avion sur 24 bits, fourni par l'OACI.

— 5 bits indiquant le type de données transmises. Cela permet de les décoder de la bonne manière, chaque type de données ayant un encodage différent.

— Les données transmises, sur 51 bits (position, altitude, vitesse, identification ...).

— 24 bits pour le Contrôle de Redondance Cyclique (CRC). Cela sert à s'assurer que le message n'a pas été altéré pendant le vol. Ce n'est cependant pas une protection cryptographique. Il permet seulement de détecter des erreurs accidentelles causées par exemple par une collision entre deux messages, ou bien par des phénomènes de réflexion sur le sol.

En ce qui concerne les données, elles sont encodées selon différents formats en fonction de leur type. Les annexes A, B, et C présentent respectivement les formats de position, d'identification, et de vitesse tels que définis par l'OACI (2008).

2.3 Objectifs de sécurité et menaces correspondantes

Il y a en sécurité informatique certains objectifs récurrents. On parle de *disponibilité*, de *confidentialité*, d'*authentification*, d'*intégrité* ou encore de *non-répudiation*. Chacun de ces objectifs a sa propre signification dans le contexte de l'ADS-B, et répond à certaines menaces largement décrites par McCallie et al. (2011), Strohmeier et al. (2015a) et Schäfer et al.

(2013). Nous allons donc nous intéresser aux principaux modèles d'attaque possibles, puis nous expliquerons quels objectifs de sécurité seront considérés dans ce mémoire.

2.3.1 Disponibilité

Le premier objectif est la disponibilité de l'ADS-B, dans le sens de la réception des messages sans interruption. En effet, il est indispensable pour la sécurité du contrôle du trafic aérien que la position des avions soit connue en permanence. La disponibilité repose ici à la fois sur le fonctionnement du transpondeur de l'avion et sur l'accès au medium de transmission.

En ce qui concerne le transpondeur, seuls les pilotes y ont accès. Excepté dans le cas où un pirate de l'air prendrait le contrôle de l'avion et couperait volontairement le transpondeur afin de passer inaperçu, il n'y a donc aucune raison que celui-ci soit désactivé. De plus, toute l'avionique passe des tests poussés avant d'être certifiée et de pouvoir être installée dans un appareil. Le risque de panne est donc limité. Enfin, le fonctionnement même du transpondeur est très restrictif. Dans le cas de l'ADS-B, il ne fait que transmettre périodiquement des messages. Les seules informations en entrée sont en provenance des capteurs, difficilement manipulables. Dans le cas du Mode S, le transpondeur répond uniquement à certains signaux précis émis par les radars. On peut donc difficilement imaginer un attaquant corrompre la mémoire du transpondeur afin de le rendre inopérant.

En ce qui concerne la disponibilité du medium de transmission, à savoir la bande de 1090 MHz, cela est plus problématique. En effet, une technique bien connue et notamment utilisée par différentes armées sur le théâtre d'opérations est le *jamming*, ou brouillage de fréquence. Elle consiste à émettre avec une grande puissance sur la fréquence que l'on veut rendre inaccessible. Il n'y a pas de contre-mesure possible, si ce n'est utiliser une autre fréquence (on pourrait par exemple transmettre temporairement sur l'UAT, mais cela nécessite un transpondeur compatible avec les deux bandes de fréquence). L'autre solution consiste tout simplement à couper l'antenne source du *jamming*, qui est en général facilement repérable en regardant la direction d'où provient le signal. Cela nécessite cependant une intervention physique, et aucune modification du protocole ne peut prévenir cette menace. Il existe enfin une variante du *jamming*, qui est le brouillage ciblé. Il consiste à se synchroniser sur un avion en particulier et à générer une interférence destructive lorsque celui-ci émet. Cela permet de rester inaperçu, puisque les contrôleurs reçoivent toujours le reste des messages. Cependant, cela est rendu très difficile par le caractère pseudo-aléatoire du moment d'envoi des messages ADS-B.

Puisque le protocole n'influence pas la disponibilité des messages (la problématique du *jamming* est propre à la communication sans fil), nous n'étudierons pas plus en détail l'objectif

de *disponibilité* par la suite. Des procédures sont déjà en place pour interagir avec les contrôleurs par radio et sur différentes fréquences en cas de défaillance du transpondeur et de non-réception des messages ADS-B.

2.3.2 Confidentialité

Dans le cadre de l'ADS-B où les messages sont diffusés par ondes radio et peuvent être reçus dans un rayon pouvant aller jusqu'à plus de 500 km, la confidentialité consiste à ce que seuls les contrôleurs soient capables d'interpréter les messages. De tierces personnes ne devraient pas être capables de pouvoir suivre en temps réel le trajet d'un avion dont le numéro d'identification est connu (et rattaché à son propriétaire).

Cet objectif de sécurité serait réalisable par un chiffrement des messages. Cependant, le fait de rendre publique la position des avions fait partie des fonctionnalités introduites avec l'ADS-B, grâce à l'ADS-B *in* et au TIS-B. La FAA va même encore plus loin, en diffusant à des partenaires industriels la position en temps réel de tous les avions dont l'information est connue, grâce à un programme nommé Aircraft Situational Display to Industry (ASDI). Cela permet à des sites comme FlightAware (2016) ou FlightRadar (2016) de rendre disponibles ces informations au grand public, notamment pour effectuer le suivi des vols commerciaux. Les seules limites sont un délai de 5 minutes pour la diffusion des informations, et la possibilité de s'inscrire sur une liste noire afin que ces informations ne soient pas diffusées (Blacker, 2013). Cependant, plusieurs de ces sites ont leur propre réseau d'antennes ADS-B, et ne sont donc pas soumis aux restrictions de l'ASDI.

Puisque l'objectif de *confidentialité* n'est pas recherché par la FAA, nous ne l'étudierons pas non plus par la suite. Il est cependant intéressant de noter qu'une piste de solution intermédiaire pourrait être l'utilisation d'un numéro d'identification anonyme, changeant à chaque vol. La correspondance avec l'identité réelle du propriétaire de l'avion serait seulement connue des contrôleurs. Les autres informations telles que la position et la vitesse de l'avion continueraient quant à elles à être diffusées normalement.

2.3.3 Non-répudiation

Un autre objectif de sécurité qui pourrait s'appliquer à l'ADS-B est la non-répudiation. Dans notre contexte, cela consisterait à pouvoir prouver qu'un avion a bel et bien suivi une trajectoire donnée, puisqu'on a reçu les messages ADS-B correspondants. Cela pourrait s'avérer utile, par exemple, pour appliquer des pénalités à un avion qui aurait survolé une zone interdite ou qui n'aurait pas suivi les directives des contrôleurs. Le pilote ne pourrait

alors pas nier les accusations portées contre lui.

Cela n'est cependant pas un objectif spécifique de la FAA. Un avion survolant une zone interdite sera directement intercepté, et un pilote ne respectant pas les directives des contrôleurs sera contacté par radio pour en connaître la raison. Nous ne poursuivrons donc pas l'objectif de *non-répudiation* par la suite.

2.3.4 Authentification et intégrité

L'objectif d'*authentification* pour l'ADS-B peut permettre de répondre à deux types de menaces. Il pourrait s'agir pour un avion de se faire passer pour un autre, en changeant son numéro d'identification. Cependant, cela n'a pas grand intérêt pour le pilote. De plus, cela ne pose pas vraiment de problème de sécurité, puisque l'objectif des contrôleurs est d'éviter les collisions, quels que soient les avions en vol.

La menace apparaît vraiment dans le cas où une personne au sol veut se faire passer pour un avion dans le ciel, tel qu'expliqué par Costin et Francillon (2012). Celle-ci peut alors utiliser un ou plusieurs drones équipés de transmetteurs ADS-B, qu'elle ferait passer pour des avions de ligne. Pour quelques centaines de dollars, un criminel peut également se procurer une antenne ainsi qu'une radio logicielle (SDR, Software Defined Radio). Avec des connaissances élémentaires en traitement du signal et sur le protocole ADS-B, il peut alors forger de faux messages, et faire apparaître aux yeux des contrôleurs des avions "fantômes". Ceci est particulièrement problématique, car les contrôleurs n'ont alors aucun moyen de distinguer ces avions fantômes des avions réels. En multipliant leur nombre dans le ciel, l'attaquant peut alors complètement perturber le travail du contrôleur. Celui-ci pourra voir apparaître de multiples risques de collisions sur ses écrans, et sera incapable de tous les traiter simultanément. Faute de recevoir des instructions fiables, le risque de collision pour un véritable avion est alors bien réel, notamment à proximité d'un aéroport où le trafic est important.

Authentifier les messages ADS-B consiste alors en deux étapes. Tous les avions doivent recevoir leur identifiant unique, émis par l'OACI (puisque l'identifiant est unique, il doit être fourni pas une entité internationale). Chaque message ADS-B est d'abord rattaché à un avion grâce à cet identifiant. La seconde étape consiste à s'assurer que seul le propriétaire de l'avion puisse utiliser son identifiant pour émettre de l'ADS-B. Ainsi, un attaquant ne possédant pas d'avion et donc pas d'identifiant ne pourra pas émettre de message et ainsi faire apparaître un avion "fantôme". Il ne pourra pas non plus envoyer des messages utilisant l'identification d'un avion réellement en vol, et modifier sa trajectoire aux yeux des contrôleurs.

L'objectif d'*intégrité* va en fait de pair avec celui d'*authentification* dans le contexte de

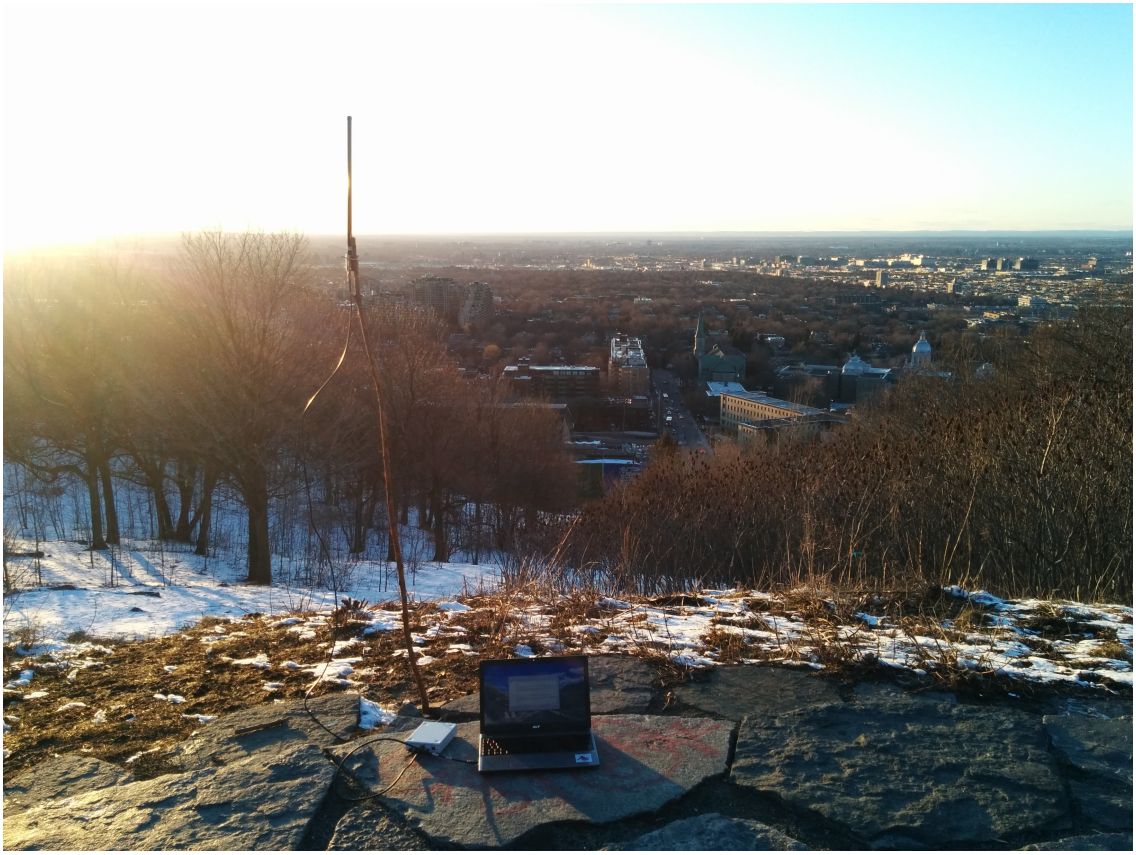


Figure 2.4 Envoi de faux messages ADS-B depuis le mont Royal. L'attaquant utilise une radio logicielle et une antenne afin de forger puis d'émettre des messages ADS-B sur la bande de fréquence de 1090 MHz.

l'ADS-B. Considérant que ce dernier objectif est atteint, un attaquant voulant perturber les services d'ATC n'a alors pas accès à un identifiant unique, et ne peut donc pas se faire passer pour un avion fantôme. En revanche, il pourrait intercepter les messages émis par un avion réel, les modifier, puis les retransmettre. Ainsi, les contrôleurs verraient bien apparaître sur leurs écrans des avions physiquement présents, mais dont la trajectoire pourrait être modifiée par l'attaquant. Dans ce cas de figure, les contrôleurs fourniraient de mauvaises indications aux pilotes. En manipulant la visualisation des trajectoires de façon réfléchie, l'attaquant pourrait ainsi facilement provoquer des collisions entre avions. Cependant, pour cet objectif de sécurité et contrairement à celui de *disponibilité*, l'utilisation d'un canal de communication sans fil est un avantage. Il est impossible d'intercepter puis de bloquer la transmission d'un message ADS-B. Le message original sera dans tous les cas diffusé. De plus, à partir du moment où le message est modifié, l'authentification ne sera de toute façon plus vérifiée.

Le risque est donc plutôt celui d'une attaque par rejeu (*replay attack*), où l'attaquant redif-

fuse un message diffusé dans le passé. Stricto sensu, l'attaque par rejeu est une usurpation d'identité, puisqu'on est alors capable d'émettre un message valide en utilisant l'identifiant unique d'un avion réel, alors que celui-ci n'est peut-être déjà plus présent dans le ciel. On a alors bel et bien fait apparaître un avion fantôme. Cependant, dans le contexte de l'ADS-B, on s'attaque aussi à l'intégrité non pas du message, mais de la trajectoire de l'avion. En effet, on peut faire croire aux contrôleurs que l'avion est toujours au-dessus de l'aéroport, ce qui pourrait légitimement être le cas par exemple sur un circuit d'attente circulaire pour l'atterrissage.

Ce sont donc ces deux objectifs conjoints d'*authentification* et d'*intégrité* qui seront traités tout au long de ce mémoire. Ce sont ceux qui présentent les plus grandes menaces s'ils ne sont pas réalisés, tout en pouvant être atteints par une évolution du protocole ADS-B.

2.4 Travaux antérieurs sur la sécurisation de l'ADS-B

De nombreux travaux ont déjà été réalisés, proposant un large éventail de méthodes pour sécuriser l'ADS-B. Chacune des différentes approches cherche à répondre à une partie des objectifs de sécurité que nous venons de présenter. Strohmeier et al. (2015a) présentent de manière très complète une grande partie de ces recherches. L'objectif ne sera donc pas ici de reprendre de manière exhaustive l'ensemble des travaux réalisés, mais plutôt de donner un aperçu des différentes manières d'aborder la problématique de la sécurisation de l'ADS-B, en gardant un œil critique sur les difficultés d'implémentation. Nous tâcherons de mettre en avant les points positifs de chaque méthode, afin de potentiellement les réutiliser au cours de notre recherche. Nous prendrons également le soin de noter ce qui nous paraît incompatible avec une mise en application réelle au sein d'une version sécurisée du protocole ADS-B, qui est l'objectif de nos travaux.

2.4.1 Vérification de la validité des données reçues

Une première catégorie de recherche a bien compris la difficulté de sécuriser le protocole ADS-B. Cela est dû à plusieurs paramètres. Tout d'abord, en aéronautique, tout bouge très lentement. En effet, l'avion n'est pas le moyen de transport le plus sûr au monde sans raison (Laffargue, 2016). Tout l'équipement d'un avion, y compris le transpondeur, doit être certifié. Cela peut prendre beaucoup de temps et nécessite de lourds investissements, qui sont bien entendu répercutés sur le prix de vente. Ainsi, l'avionique est généralement conçue pour avoir une très longue durée de vie et n'est souvent remplacée qu'en cas de panne ou de nouvelles réglementations. Les instances gouvernementales comme la FAA essayent en conséquence

de limiter ces dernières, afin de ne pas mécontenter les pilotes qui ne veulent pas avoir à investir sans cesse des milliers de dollars pour être autorisés à voler. La mise en place du plan *NextGen* dont la pierre angulaire est l'ADS-B fait donc figure d'exception, car il est nécessaire de remplacer son transpondeur pour le rendre compatible avec ce nouveau protocole. Alors que le processus de renouvellement des transpondeurs est déjà bien avancé, il n'est donc pas concevable qu'une évolution sécurisée de l'ADS-B impose à nouveau son remplacement à court ou moyen terme. L'idée avancée est donc de tout simplement ne pas toucher à la version actuelle de l'ADS-B. On accepte alors que les données que l'on reçoit peuvent être erronées, et de les vérifier d'une autre manière. Comme la principale information émise est la position de l'avion (c'est celle qui est primordiale pour le contrôle du trafic aérien), de nombreux travaux proposent différents moyens de s'assurer qu'elle est valide.

2.4.1.1 Multilatération

La première solution, très développée dans la littérature, consiste à utiliser la technique de la multilatération (MLAT). L'idée est simple, et consiste en une extension des techniques de triangulation et de trilatération utilisées depuis des siècles notamment pour la navigation. Cependant, contrairement à ces deux méthodes qui utilisent respectivement des mesures absolues d'angles et de distances pour se repérer par rapport à des points de référence dont les positions sont connues, la multilatération utilise des mesures relatives. En effet, la position des antennes ADS-B au sol est fixe et connue. Cependant, la distance d'un avion à chacune de ces antennes, elle, ne l'est pas. La seule information que l'on est capable d'obtenir est le moment auquel un message a été reçu au niveau d'une antenne déterminée (Time Of Arrival, TOA). En synchronisant le temps de référence entre deux antennes proches ayant reçu le même message ADS-B, on est capable de mesurer la différence de temps d'arrivée entre ces deux antennes (Time Difference Of Arrival, TDOA). À partir de cette mesure, on est alors capable de tracer un hyperboloïde sur laquelle doit se trouver l'avion. Chaque antenne additionnelle nous apporte ensuite un nouveau TDOA et une nouvelle hyperboloïde. À partir de quatre antennes et trois hyperboloïdes, l'intersection de celles-ci est un point et on a obtenu la position de l'avion (plus le nombre d'antennes est important, meilleure est la précision).

Le principal avantage de la multilatération est qu'elle repose sur l'infrastructure existante : les avions émettent déjà un signal de manière périodique, et on peut se servir du réseau d'antennes ADS-B installées pour calculer les TDOA. On peut au besoin rajouter un certain nombre d'antennes au sol pour augmenter la précision, car leur coût est bien inférieur à celui d'un radar. Galati et al. (2005) proposent des algorithmes permettant d'implémenter les

principes de la multilatération dans le cadre de l'ADS-B, et en décrivent les performances. Ces résultats, confirmés par Smith et al. (2006), montrent que la technique de la multilatération fourni des taux de rafraichissement et une précision bien supérieure à celle des radars lorsque les avions sont éloignés. En effet, le radar mesure l'angle de réception du signal, et l'erreur est donc proportionnelle à la distance.

La multilatération apporte cependant aussi certains problèmes, comme expliqué par Strohmeier et al. (2015b). Si l'éloignement de l'avion influe peu sur la précision du positionnement, celle-ci est cependant très susceptible à des erreurs même minimes sur les temps d'arrivée. Le rapport entre l'erreur sur l'estimation de positionnement et l'erreur sur le temps d'arrivée (TOA) est nommé Geometric Dilution Of Precision (GDOP). Plus le GDOP est grand, plus la position calculée de l'avion risque donc d'être imprécise. La valeur du GDOP dépend de la position des antennes au sol ; il ne faut notamment pas qu'elles soient trop rapprochées. Un cas particulier du GDOP est le Vertical Dilution Of Precision (VDOP). Les antennes aux sols sont généralement situées sur un même plan, car le relief au sol est limité. Leurs différences de coordonnée sur l'axe vertical sont donc très faibles, et le VDOP est grand. En conséquence, il est très difficile d'obtenir l'altitude des avions avec une grande précision grâce à la multilatération.

On pourrait alors penser placer des antennes en montagne afin de corriger ce problème. Cependant, la MLAT est très susceptible au problème du multi-trajet, qui est le fait que le message soit réfléchi sur l' ionosphère ou un obstacle quelconque. Il se propage en conséquence par plusieurs chemins, et est en fin de compte reçu plusieurs fois par l'antenne à des instants légèrement différés. Or, comme on vient de le voir, même un léger décalage à la réception peut entraîner une grande erreur de positionnement à cause du GDOP. Et le fait de placer une antenne en montagne augmente la probabilité que le problème du multi-trajet se produise, par réflexion sur les parois rocheuses.

Le placement des antennes est donc un sujet très sensible avec la multilatération. Si au minimum quatre antennes distinctes n'ont pas reçu le signal ADS-B, la vérification du positionnement est impossible. La bande des 1090 MHz est déjà saturée et la probabilité de collision entre deux messages n'est pas négligeable, comme nous le détaillerons au chapitre 5. En ajoutant à cela le problème du multi-chemin qui peut aussi provoquer une auto-collision si le décalage à la réception est inférieur à la durée du message, la probabilité que la réception ait réussi pour quatre antennes simultanément peut devenir très faible. Sur cinquante millions de messages analysés et avec un réseau de 8 antennes, Strohmeier et al. (2015b) montrent que seulement 4% d'entre eux ont été correctement reçus par au minimum quatre antennes. En prenant en compte le GDOP, il ne reste que 0.36% des messages dont les données de

position ont pu être vérifiées de manière fiable. Pour corriger cela, Strohmeier et al. (2015b) proposent d'utiliser une méthode statistique. Un quadrillage est créé, divisant l'espace en différentes sections suffisamment petites. Lors d'une phase d'apprentissage, on calcule pour chaque section de la grille une distribution statistique des valeurs des TDOA à partir des données enregistrées dans le passé. Lorsqu'un avion prétend être à une position donnée, on va mesurer la distribution des TDOA pour plusieurs de ses messages ADS-B. En la comparant à la distribution précalculée, on est alors capable de vérifier si l'avion se situe réellement dans la section où il prétend être.

Cette dernière solution reste tout de même limitée. Il faut en premier lieu régulièrement actualiser les tables de valeurs. De plus, on doit attendre d'avoir reçu plusieurs messages avant de pouvoir comparer les distributions statistiques. Réduire le maillage du quadrillage et augmenter la précision nécessite toujours d'augmenter le nombre d'antennes et va également grandement augmenter le temps de calcul. De plus, la multilatération nécessitera toujours de relier les différentes antennes entre elles et de synchroniser leurs horloges de manière très précise, ce qui est difficile. Enfin, la multilatération n'est pas possible au-dessus des grands espaces vierges comme la baie d'Hudson, où il serait très coûteux d'avoir un fort maillage d'antennes ADS-B, et n'est pas applicable non plus au-dessus des océans où on ne trouve pas d'antenne.

La technique de la multilatération pourrait donc être utilisée pour sécuriser l'ADS-B, mais seulement là où le maillage d'antennes est très dense. Cependant, cela a alors un coût d'infrastructure important, et c'est d'ailleurs la raison pour laquelle la MLAT n'a pas été choisie comme solution à grande échelle pour remplacer les radars. De plus, seule la position de l'avion pourrait être vérifiée. Nous avons vu que pour l'altitude l'information ne serait pas fiable, et toutes les données annexes comme l'identification de l'avion ne pourraient pas être confirmées. Il faut donc trouver une autre solution dans la majorité des cas.

2.4.1.2 Vérification de groupe

Plutôt que d'utiliser le principe de la multilatération avec des antennes au sol, ce qui demande un fort investissement, Sampigethaya (2010) propose de regrouper les avions par quatre ou plus afin qu'ils procèdent eux-mêmes à la vérification des informations d'un avion extérieur au groupe par multilatération. Cependant, cela pose toute une série de problèmes. Il faut tout d'abord qu'il y ait suffisamment d'avions à proximité pour qu'ils puissent se regrouper. Selon l'article, cette condition est remplie dans 91% des cas aux États-Unis, mais qu'en est-il ailleurs ? De plus, il est très facile pour un attaquant de générer plusieurs avions fantômes autour d'un avion réel, qui rejoindront son groupe afin de fausser la vérification. Enfin, tout

ce processus de validation groupée demande un nombre important d'échanges de données bidirectionnels, qui devra se faire sur un canal dédié. Les auteurs proposent l'utilisation du standard L-band Digital Aeronautical Communications System (LDACS) en cours de développement par EUROCONTROL. Cependant, il est conçu pour la communication air-sol et non air-air. De plus, cela nécessiterait un nouvel équipement à intégrer dans les avions, ce qui est tout à fait contraire à nos objectifs.

2.4.1.3 Filtre de Kalman

Une autre solution discutée dans la littérature est l'utilisation d'un filtre de Kalman. À partir d'une série de mesures antérieures, celui-ci est capable de déduire la dynamique du système, et ainsi de prédire les valeurs futures. Krozel et al. (2004) proposent donc d'utiliser le filtre de Kalman pour estimer la trajectoire future d'un avion. Cependant, le pilote peut choisir à un moment donné de changer de trajectoire pour, par exemple, contourner une zone orageuse. L'idée est alors de se baser sur un type de message ADS-B particulier, qualifié "d'intention", et qui est justement destiné à indiquer que l'avion va volontairement changer de trajectoire. Celle calculée par le filtre de Kalman serait alors corrigée en conséquence.

Cette méthode peut paraître judicieuse, mais elle a toutefois de nombreux problèmes. Elle est en réalité conçue pour corriger du bruit ou des erreurs aléatoires, et non pas une modification volontaire des données de la part d'un attaquant. En effet, Kovell et al. (2012) évoquent l'attaque qualifiée de "frog-boiling", soit de la "grenouille ébouillantée". Placez une grenouille dans une marmite d'eau bouillante, et celle-ci sautera pour s'en échapper. À l'inverse, placez-la dans une marmite d'eau froide, que vous porterez lentement à ébullition. La grenouille ne se rendra alors pas compte du changement progressif de température, et restera dans la marmite jusqu'à sa mort. Il en est de même pour le filtre de Kalman. Si l'attaquant modifie suffisamment progressivement la trajectoire d'un avion, cela passera inaperçu.

De plus, dans le cas d'un nouvel avion apparaissant dans le ciel, il n'est pas possible de distinguer s'il est réel ou si c'est un avion fantôme, puisqu'on ne dispose pas de données antérieures. Enfin, en ce qui concerne les "intentions", celles-ci ne sont actuellement presque jamais transmises par ADS-B. Dès qu'un avion changera de trajectoire, le filtre de Kalman produira donc un faux positif d'attaque. De toute manière, l'utilisation des "intentions" ne serait pas fiable, puisqu'un attaquant pourrait très bien les modifier pour faire croire qu'un changement de trajectoire est réel alors que ce n'est pas le cas. L'utilisation du filtre de Kalman n'est donc pas adaptée à la sécurisation de l'ADS-B.

2.4.1.4 Contraintes de distance

Strohmeier et al. (2015a) proposent l'idée de calculer des contraintes de distance d'un avion par rapport à des antennes au sol. Si l'on envoie un challenge à un avion, il ne peut pas répondre plus rapidement que le temps d'un aller-retour de l'information à la vitesse de la lumière entre l'antenne et l'avion. Un attaquant est donc capable de rallonger le temps de réponse au challenge, mais pas de le diminuer. Ainsi, on est capable de calculer une borne supérieure de la distance entre l'avion et l'antenne. En croisant les bornes calculées entre différentes antennes, on est alors capable de vérifier si la position annoncée pour un avion donné est plausible. On peut également se baser sur la vitesse maximale d'un avion pour calculer la distance maximale entre deux positions successives.

L'idée est astucieuse et, même si peu précise, pourrait donner une première idée de la validité des données de position ADS-B. Les transpondeurs actuels étant déjà capables de répondre aux interrogations des SSR, il n'y aurait pas de problème de compatibilité matérielle. Cependant cette solution ne permet pas de détecter un attaquant qui ferait apparaître des avions au-dessus de lui, puisque les contraintes de distance seraient respectées. De plus, plusieurs antennes émettrices bien placées peuvent facilement rendre possible une attaque à grande échelle. Cette solution n'est donc pas suffisante pour sécuriser l'ADS-B.

2.4.1.5 Puissance du signal reçu

Strohmeier et al. (2015a) proposent également de mesurer en permanence la puissance du signal reçu pour chaque message. Si une série de messages est reçue de la part d'un même avion avec exactement la même puissance, c'est probablement que le signal est émis depuis une antenne au sol, car un avion ne peut pas être en vol stationnaire. Cependant, l'attaquant peut très bien faire varier la puissance d'émission de son antenne avec le temps. Une autre solution serait d'adapter la méthode de distribution statistique des TDOA de Strohmeier et al. (2015b) pour la puissance des signaux. Cependant, celle-ci peut grandement varier en fonction des conditions météo, et la méthode serait donc très peu fiable.

2.4.1.6 Vérification à base d'ontologies

Une dernière possibilité également abordée par Strohmeier et al. (2015a) est de se baser sur les lois de la physique, les règles aéronautiques et les capacités techniques des avions pour détecter si les messages reçus sont plausibles ou non. Nous proposons de rentrer toutes ces informations dans une base de données ontologique, qui est capable de représenter des concepts et leurs relations à haut niveau. On pourra par exemple y retrouver le plan de vol

de l'avion, le type d'appareil avec sa vitesse de croisière typique, son altitude maximale de vol ou encore les cartes de navigation comprenant les différents couloirs aériens et les zones interdites de vol. À travers un langage de requêtes ontologiques tel que SPARQL (W3C, 2008), nous pourrons alors déterminer si toutes les caractéristiques de vol en condition normale sont respectées, et donc si oui ou non nous sommes potentiellement en présence d'une attaque.

Cette méthode paraît prometteuse et mériterait d'être plus amplement développée au sein d'un travail de recherche spécifique. Cependant, elle a seulement pour objectif de détecter des comportements étranges de la part de certains avions, et donc de complexifier la tâche d'un attaquant. Celui-ci pourra toujours s'appliquer à respecter toutes les caractéristiques d'un vol légitime, et il ne sera alors pas détecté. Une autre solution de sécurisation de l'ADS-B est donc nécessaire si l'on veut prévenir tout risque d'attaque.

2.4.2 Sécurisation du protocole avec des solutions cryptographiques

La validation des données de positionnement par les différentes méthodes précédemment énoncées peut s'avérer efficace dans certains cas particuliers, mais n'est pas suffisante pour sécuriser l'ADS-B de manière plus globale. En particulier, même dans le cas où l'on serait capable de garantir qu'un message est bel et bien envoyé depuis la position prétendue, l'objectif d'authentification ne serait pas assuré. Rien ne garantit que l'émetteur soit un avion ; il pourrait s'agir d'un drone commandé par un attaquant pour simuler un risque de collision. Il est donc nécessaire d'étudier les possibilités d'évolution du protocole ADS-B afin d'y intégrer une solution cryptographique permettant d'atteindre nos objectifs de sécurité, à savoir l'authentification et la vérification de l'intégrité des messages.

2.4.2.1 Chiffrement

Un premier procédé cryptographique commun est le chiffrement. L'objectif principal est bien entendu de rendre le message transmis incompréhensible si l'on ne possède pas la clé de déchiffrement. Les objectifs d'authentification et d'intégrité peuvent cependant également être atteints. Soit Alice et Bob partageant une clé symétrique, connue d'eux seuls. Alice transmet un message ADS-B à Bob. Oscar, un attaquant, l'intercepte et tente de le modifier en inversant certains bits. À la réception du message, Bob déchiffre celui-ci, puis vérifie le CRC. Puisque le message a été modifié par Oscar, ceux-ci ne sont pas valides. Bob sait alors que le message ne provient pas d'Alice, a été modifié ou a subi des erreurs pendant la transmission.

Le chiffrement pose cependant plusieurs problèmes. Tout d'abord, nous avons vu que nous

ne recherchions pas l'objectif de confidentialité, et que celui-ci était même indésirable, car le fait que les données ADS-B soient diffusées publiquement est considéré comme une fonctionnalité du protocole. De plus, le chiffrement casse le format du message tel que présenté à la figure 2.3, et donc la rétrocompatibilité. Pour corriger ce problème, Finke et al. (2013) proposent d'utiliser la technique de Format-Preserving Encryption (FPE), qui consiste à faire en sorte que le format du message avant et après le chiffrement reste le même. Pour cela, ils mettent en application l'algorithme symétrique FFX de la National Institute of Standards and Technology (NIST), basé sur l'Advanced Encryption Standard (AES). Cependant, même en gardant le bon format, la solution ne sera pas rétrocompatible puisque les transpondeurs existants seront certes capables de recevoir les messages, mais pas d'en déchiffrer les informations. Cette solution n'est donc pas satisfaisante pour sécuriser l'ADS-B.

2.4.2.2 Code d'authentification

Une autre solution cryptographique est celle des codes d'authentification, ou Message Authentication Code (MAC). Celle-ci permet de répondre simultanément aux objectifs d'authentification et d'intégrité. Le principe consiste à ce que l'émetteur, Alice, produise ce que l'on appelle un MAC du message. Il est caractéristique de celui-ci, et ne peut être généré qu'à partir d'une clé secrète K. Une fois le MAC généré, il faut le transmettre avec le message. Bob, qui reçoit le message, doit également connaître la clé secrète k (c'est une clé symétrique). Il est alors capable de générer lui-même le MAC à partir du message reçu et de K. Il compare le MAC qu'il a calculé à celui transmis avec le message. S'ils sont identiques, le message provient bien d'Alice et n'a pas été modifié lors de la transmission. En effet, un attaquant, Oscar, ne connaît pas K. S'il altère le contenu du message, il ne pourra donc pas recalculer le MAC correspondant et Bob se rendra compte de l'attaque.

Le MAC est calculé à partir d'une fonction de hachage h que l'on applique à une combinaison du message m et d'une clé secrète K. On pourrait par exemple avoir :

$$MAC_K(m) = h(K \| m) \tag{2.1}$$

Cependant, ce type de MAC est vulnérable à plusieurs types d'attaques, comme celle de l'extension (Bellare et al., 1996). Le problème apparaît lors de l'utilisation de certaines fonctions de hachage traditionnelles comme MD5, SHA-1 ou SHA-2, basées sur la construction de Merkle-Damgård. À partir du MAC généré, on est capable de reproduire l'état interne de la fonction de hachage. Ainsi, on peut à partir de cet état (qui sert alors le nouvel état

initial) continuer à exécuter la fonction de hachage sur de nouvelles données m'. On obtient alors le MAC valide $MAC_K(m\|m')$ sans pour autant avoir eu besoin de connaître K.

On utilise donc généralement plutôt un keyed-Hash Message Authentication Code (HMAC) (Krawczyk et al., 1997), qui est un type de MAC non vulnérable à ce type d'attaque :

$$HMAC_K(m) = h\Big((K' \oplus opad) \parallel h\big((K' \oplus ipad)\|m\big)\Big) \tag{2.2}$$

Si l'on utilise une fonction de hachage qui travaille sur des blocs de B octets,

- K' représente la clé K que l'on a complétée par des 0 jusqu'à atteindre B octets
- opad est l'octet 0x36 répété B fois
- ipad est l'octet 0x5C répété B fois

La taille des MAC produits dépend de la fonction de hachage utilisée. Elle est par exemple de 128 bits pour MD5, 160 bits pour SHA-1, et est variable pour SHA-2 (256 bits pour SHA-256, 512 bits pour SHA-512, ...).

Kacem et al. (2015a) étudient les différentes possibilités de transmission du MAC avec le message ADS-B correspondant. Ils évoquent la possibilité de modifier le format des messages, mais cela casserait la rétrocompatibilité que l'on recherche. Ils proposent également d'envoyer des messages supplémentaires, avec des types d'information qui n'occuperaient pas les 51 bits de données disponibles. L'espace libre serait alors utilisé pour transmettre le MAC. Cependant, cela n'est pas réaliste. Dans les rares cas où quelques bits restent disponibles, ceux-ci sont marqués comme "réservés" et pourraient très bien servir dans une prochaine version du protocole. De plus, cela utiliserait beaucoup de bande passante de manière inutile, puisque seule une infime partie du message serait utilisée pour transmettre le MAC.

Enfin, une dernière solution avancée serait de tronquer la taille du MAC à 24 bits, et de l'envoyer à la place du CRC. Cependant, cela poserait là aussi un problème de rétrocompatibilité puisque les anciens transpondeurs trouveraient le CRC erroné, et supprimeraient le message. L'article suggère de plus que tronquer la taille du MAC augmenterait les risques de collision pour celui-ci, c'est-à-dire la probabilité qu'un message différent produise le même MAC. Cela pourrait ainsi selon les auteurs affecter la sécurité de la solution. Nous discuterons ce point en détail au chapitre 5.2. Pour contourner le potentiel problème, il est proposé de rassembler les messages par groupes. Le MAC ne sera plus calculé pour chaque message, mais pour la concaténation de l'ensemble des messages du groupe. Il sera ensuite envoyé par morceaux de 24 bits à la place des CRC de chacun des messages. Par exemple, si l'on utilise une fonction de hachage produisant des MACs de 128 bits, les messages seront regroupés par 6, car c'est la

taille minimum du groupe pour être capable de transmettre tous les bits du MAC. Cependant, le problème de la rétrocompatibilité est toujours présent. De plus, puisqu'il est nécessaire de connaître le contenu de tous les messages avant de pouvoir calculer le MAC, cela nécessite de retarder l'envoi de tout le groupement au moment où le dernier message sera généré. Avoir un certain délai pour la vérification de l'authenticité et de l'intégrité des messages n'est pas forcément problématique, comme nous l'expliquerons au chapitre 4. Cependant, les messages doivent être envoyés en temps réel, car c'est là que réside tout l'intérêt de l'ADS-B. On veut pouvoir connaître la nouvelle position de l'avion deux fois par seconde, et non pas ses six dernières positions toutes les trois secondes. Cette approche n'est donc pas envisageable.

2.4.2.3 Gestion et distribution de clés symétriques

Les deux solutions cryptographiques que nous venons d'étudier utilisaient des clés symétriques, c'est-à-dire des clés qui doivent être partagées entre l'émetteur du message et tous les récepteurs. Si Alice et Bob sont les deux seuls à communiquer et à partager une clé symétrique, il n'y a pas de problème. Puisqu'à part lui, Alice est la seule personne à connaître la clé, Bob sera certain qu'un message chiffré ou ayant un MAC produit avec cette clé proviendra forcément d'Alice. En effet, Bob fait confiance à Alice pour ne pas diffuser la clé. Mais qu'advient-il si Oscar se joint à la conversation ? Oscar est un pilote, il est en droit de recevoir la clé pour pouvoir authentifier lui aussi les messages ADS-B. Il n'est malheureusement pas possible de s'assurer que tous les pilotes ont de bonnes intentions. Il se trouve qu'en réalité Oscar est un attaquant, et il va utiliser cette clé pour générer des avions fantômes. L'utilisation de clés symétriques cause donc de gros problèmes pour répondre à l'objectif d'authentification.

Finke et al. (2013) proposent de ne diffuser la clé qu'à un nombre réduit de pilotes dont on aurait vérifié les antécédents. Ils s'appuient pour cela sur le mode 4 du système d'identification (IFF) sécurisé utilisé par l'Organisation du Traité de l'Atlantique Nord (OTAN), et qui utilise des clés symétriques. Cependant, le risque de fuite des clés est bien plus faible dans l'armée que pour le civil. De plus, dans le mode militaire, les clés sont changées tous les jours au cas où une fuite de la clé se serait tout de même produite malgré toutes les précautions prises. Il n'est pas concevable de demander à un pilote civil de mettre à jour la clé quotidiennement. Dans tous les cas, on ne veut pas réduire la capacité d'authentification à un nombre réduit de pilotes.

Wesson et al. (2014) recommandent l'idée d'enregistrer la clé dans une mémoire sécurisée et inviolable au sein du transpondeur. Cela réduit les risques, mais est loin d'être infaillible. Tout d'abord, tous les fabricants d'avionique devront posséder la clé, et ils peuvent être à

l'origine de la fuite. De plus, une erreur de conception du transpondeur pourrait finir par mener à la découverte de la clé. Il faudrait alors mettre à jour la totalité d'entre eux, ce qui serait complexe et nécessiterait beaucoup de temps et de moyens. Enfin, un attaquant pourrait tout simplement récupérer des transpondeurs usagers ou en voler afin d'être capable de transmettre des messages avec ceux-ci.

Un brevet a été déposé par la société *Saab Sensis Corporation*, et propose une alternative (Viggiano et al., 2010). Plutôt que d'utiliser une unique clé symétrique partagée par tous les avions, chacun d'entre eux aurait sa propre clé. Celle-ci ne serait connue que par l'avion lui-même et par les contrôleurs aériens, qui les regrouperaient au sein d'une large base de données. Les avions ne seraient donc pas capables de s'authentifier directement entre eux, mais les contrôleurs, eux, seraient capables d'authentifier tous les avions. Ils utiliseraient alors un service de diffusion, le TIS-B existant ou bien un nouveau créé spécifiquement, pour transmettre la liste de tous les avions authentifiés aux alentours. C'est une solution très intéressante, mais si la base de données des clés se trouvait diffusée, il faudrait alors remplacer les clés de tous les avions. Ceci serait, comme expliqué précédemment, un processus très complexe. On peut supposer qu'une fuite serait peu probable au sein d'une instance d'ATC comme la FAA. Cependant, étendre la solution d'authentification internationalement supposerait de partager la base de données avec chaque pays membre de l'OACI. Une si large diffusion rend beaucoup plus probable le risque de fuite. Enfin, cette solution n'est pas compatible avec les zones géographiques où la couverture d'antennes au sol ne permet pas de fournir le service de diffusion, comme les vastes zones inoccupées ou les océans. On pourrait penser le fournir à travers la constellation de satellites *Aireon*, déjà chargée de recevoir les messages ADS-B dans ces zones, mais ces satellites ne sont pas conçus pour émettre un signal de type TIS-B. En conclusion, *Saab Sensis Corporation* est une entreprise sérieuse du domaine de l'ATC, ayant déjà eu de nombreux contrats avec la FAA. Ils fournissent notamment des antennes ADS-B capables d'effectuer de la multilatération et de partiellement vérifier les données de positions, comme expliqué précédemment. Leur solution semble efficace, et pourrait être mise en application relativement rapidement du fait de leur expertise et de leur brevet. Cependant, ce même brevet peut freiner son adoption au niveau international et sa normalisation par l'OACI. Elle présente également certaines limitations que nous avons énoncées, ce qui nous motive à continuer notre recherche d'une solution plus universelle.

2.4.2.4 Signature numérique

La signature numérique, tout comme les MAC, permet de répondre à la fois aux objectifs d'authentification et d'intégrité. De la même manière que ces derniers, elle consiste à rajouter un certain nombre de bits à la fin du message. Ceux-ci sont calculés par l'émetteur à partir d'une clé. Ils servent à vérifier qu'il est bel et bien celui qui a envoyé le message, et que ce dernier n'a pas été altéré en cours de transmission. Cependant, il existe une différence majeure entre les MAC et la signature électronique. Alors que les premiers utilisent une clé symétrique, servant à la fois à générer le code d'authentification et à le vérifier, la seconde utilise deux clés distinctes, qualifiées d'asymétriques. Cela signifie que chaque avion possède une clé privée K_{pr} connue de lui seul. À cette clé privée est associée une clé publique K_{pb}, qui, elle, sera diffusée à tous les autres avions et aux contrôleurs. Le processus est alors le suivant :

— Bob, l'avion qui envoie le message ADS-B, choisit une fonction de hachage cryptographique sans faille de sécurité connue h. Il peut par exemple s'agir d'une des variantes de SHA-2.

— Bob applique h à son message m. Il obtient alors un hash.

— Bob choisit un protocole de cryptographie asymétrique connu. Il peut s'agir de El Gamal, basé sur le problème du logarithme discret, de RSA, basé sur la factorisation du produit de deux grands nombres premiers, ou encore de cryptographie sur courbes elliptiques. Quelle que soit la méthode choisie, il existe une fonction de chiffrement c utilisant K_{pr} et une fonction de déchiffrement d utilisant K_{pb}.

— Bob peut alors calculer la signature S de son message :

$$S_{K_{pr}}(m) = \text{Sign}\left(K_{pr}, h(m)\right) \tag{2.3}$$

— Bob attache la signature à son message, puis le diffuse. Alice, un autre avion, reçoit le message ADS-B de Bob et veut vérifier qu'il en est bien l'auteur.

— Alice connaît la clé publique de Bob. Elle déchiffre donc la signature attachée au message :

$$\text{Verif}\left(K_{pb}, S_{K_{pr}}(m)\right) \;=\; h(m) \tag{2.4}$$

— Alice applique la même fonction de hachage que Bob sur le message. Si le résultat qu'elle obtient est le même *hash* qu'elle a obtenu en déchiffrant la signature, alors Bob est bel et bien l'émetteur et le message n'a pas été altéré.

L'avantage de cette solution par rapport aux MAC est que la clé publique peut être diffusée sans risque à n'importe qui souhaitant authentifier les messages ADS-B. En effet, cette clé ne permet que la vérification des signatures, mais pas leur création. La posséder ne permet donc pas de se faire passer pour un autre avion et de créer des avions fantômes.

Cependant, cela vient avec un inconvénient. Lorsque l'on utilisait les MAC, il était possible de tronquer leur taille afin de limiter la bande passante utilisée. Cela n'est pas possible avec la signature électronique, car il s'agit d'un processus de chiffrement du *hash* du message. Si l'on tronque la signature, il ne sera plus possible de la déchiffrer, et donc d'authentifier le message. Il est ainsi important de choisir une méthode de chiffrement asymétrique possédant un niveau de sécurité suffisant tout en produisant la signature la plus petite possible.

Wesson et al. (2014) proposent d'utiliser ECDSA (Elliptic Curve Digital Signature Algorithm), qui est une variante du *Digital Signature Algorithm* (DSA) fonctionnant avec des courbes elliptiques. Selon eux, pour un même niveau de sécurité, ECDSA produit des signatures plus courtes que DSA. Cependant, cela est incorrect, car les deux méthodes produisent en réalité des signatures d'une taille équivalente à quatre fois le niveau de sécurité recherché, calculé en nombre de bits (un niveau de sécurité de n bits signifie qu'il faut au maximum 2^n opérations pour qu'un attaquant puisse découvrir la clé privée) (Lenstra, 2006).

Les auteurs étudient ensuite l'influence de l'ajout de la signature sur la capacité du canal de 1090 MHz en termes du nombre d'avions pouvant émettre simultanément de l'ADS-B. Ils étudient le cas d'utilisation d'une signature ECDSA de 448 bits ajoutée à la fin du message, possédant donc un niveau de sécurité de 128 bits. Cela ne sera suffisant que jusqu'en 2030 selon la *National Institute of Standards and Technology* (NIST) (Barker, 2016). Cependant, leur simulation, basée sur une distribution de Poisson pour la répartition des messages dans le temps, montre que cela réduit déjà drastiquement le nombre d'avions pouvant émettre de l'ADS-B dans un rayon d'environ 300 km. Le nombre passe en effet de 350 pour le cas sans authentification à 80 avec la signature ECDSA de 448 bits, pour une probabilité de réception réussie de 99.5 % (probabilité considérée satisfaisante par Boisvert et Orlando (1993)). Une optimisation est proposée, consistant à utiliser les 56 bits de données disponibles dans un message ADS-B pour envoyer la signature par morceaux. Il faut alors ajouter 8 messages de signature pour un message de données ADS-B traditionnel. Dans ce cas de figure, la capacité du canal est réduite à 190 avions. Dans tous les cas, étant donné qu'il faudrait augmenter la taille de la signature pour la rendre sûre au-delà de 2030, l'utilisation de la signature électronique paraît inappropriée du fait de sa trop grande utilisation de bande passante. Cela limite en effet significativement le nombre d'avions pouvant émettre de l'ADS-B simultanément dans une région donnée, alors que NextGen a pour objectif d'augmenter ce nombre.

2.4.2.5 Infrastructure à clés publiques

L'utilisation des signatures électroniques et donc de la cryptographie asymétrique requiert un mécanisme de distribution des clés publiques. Or, les clés de tous les avions ne peuvent pas être stockées au sein du transpondeur, pour des raisons de stockage et de mise à jour. Pan et al. (2012) proposent de télécharger auprès de l'ATC avant le décollage toutes les clés qui seront nécessaires pour la durée du vol. Cependant, il n'est pas réaliste de penser pouvoir obtenir une bonne estimation de tous les avions qui seront croisés pendant la durée du vol. En effet, un pilote peut choisir de voler en IFR (Instrument Flight Rules), c'est-à-dire aux instruments. Il se sert alors de ses différents équipements - comme son compas, son altimètre ou encore son horizon artificiel - pour se diriger. Dans ce cas-ci, il a l'obligation de remplir un plan de vol, et l'on peut potentiellement prédire si son avion croisera notre route. Mais il n'y a pas de délai minimum pour la publication du plan de vol avant le décollage. Si un pilote publie son plan de vol au dernier moment, après que nous aurons nous-mêmes décollé, alors nous n'aurons pas téléchargé sa clé. De plus, il est possible de voler en VFR (Visual Flight Rules), c'est-à-dire en visuel. Dans ce cas-ci, il n'est pas obligatoire de remplir un plan de vol. Il n'est donc pas possible de savoir *a priori* quels seront les avions volant en VFR dont nous aurons besoin de la clé. Enfin, même lorsque les plans de vol sont connus à l'avance, il y a souvent des changements de route en cours de vol. Cela nécessiterait donc de télécharger de nouvelles clés.

Dans ce dernier cas, les auteurs proposent de créer un nouveau format de message Mode S, qui servirait spécifiquement à l'émission des clés. Il n'y a alors pas de problème de rétrocompatibilité, puisque les anciens transpondeurs ignoreraient simplement ce type de message et fonctionneraient en mode non sécurisé. Puisque l'on transmet les messages à travers la bande de fréquence de 1090 MHz, il faut cependant économiser au maximum la bande passante. L'argument d'utiliser ECDSA plutôt que DSA devient alors pertinent, puisque la taille des clés est effectivement inférieure dans le premier cas (contrairement à la taille des signatures qui elle est identique).

Enfin, la distribution des clés publiques en cours de vol n'est pas suffisante. En effet, comment s'assurer qu'une clé publique que l'on reçoit provient bien d'un avion légitime et non pas d'un attaquant qui voudrait authentifier son avion fantôme ? Pour cela, il est nécessaire de mettre en place ce que l'on appelle une ICP, ou Public Key Infrastructure (PKI). Cela signifie qu'un certain nombre d'autorités reconnues et dignes de confiance (on les appelle Certificate Authority ou CA) seront chargées de *certifier* les clés publiques des différents avions. Le fait qu'une autorité *certifie* la clé publique d'un avion donné signifie qu'elle a fait les vérifications nécessaires pour s'assurer que cet avion était bien légitime, et donc qu'il est

autorisé à transmettre de l'ADS-B. On pourra donc faire confiance aux données émises par un avion possédant un tel certificat.

Une infrastructure à clé publique fonctionne généralement sur plusieurs niveaux. Une ou plusieurs autorités qualifiées de *racine*s sont reconnues comme étant dignes de confiance. Toute la chaîne de certification repose sur cette *confiance* en l'autorité racine, puisqu'elle est en haut de l'échelle, et n'est elle-même certifiée par personne. Ces autorités racines certifient d'autres CA qu'elles considèrent comme dignes de confiance, et ces dernières peuvent elles-mêmes à leur tour certifier encore d'autres CA. Les autorités en bout de chaîne sont enfin chargées de certifier les clés publiques des avions. Costin et Francillon (2012) proposent une ICP à seulement deux niveaux. Chaque autorité responsable du transport aérien à un niveau national (par exemple la FAA ou NAV CANADA, comme vu en 2.1) serait considérée comme une autorité racine. Lors de la maintenance d'un transpondeur, la liste de toutes les autorités racines serait mise à jour à l'intérieur de celui-ci. En vol, les avions transmettent le certificat fourni par l'autorité racine de leur pays d'immatriculation en même temps que leur clé publique. On peut authentifier cette dernière à partir du certificat et de la clé publique de l'autorité racine enregistrée dans le transpondeur.

Pan et al. (2012) proposent une variante pour la structure de la ICP. Un avion reconnaîtrait une seule autorité racine, à savoir l'autorité responsable du transport aérien dans son pays d'immatriculation. Cette autorité reconnaîtrait elle-même comme sous-autorités celles des nations qu'elle considère comme dignes de confiance. Ainsi, si un état en particulier ne joue pas bien son rôle de vérification et certifie des clés pour des avions inexistants (cela pourrait par exemple être le cas dans certains pays où la corruption est courante), les autres états auraient la possibilité de ne pas lui accorder de certificat. Nous proposerons nous-mêmes une structure de ICP légèrement différente en 4.2.1.

2.4.2.6 Signature basée sur l'identité

Nous venons de voir que la transmission des clés publiques et de leurs certificats associés en cours de vol utilisait une certaine quantité de bande passante, alors que celle-ci est rare et précieuse. Baek et al. (2013) et Haomiao et al. (2013) proposent une méthode astucieuse pour contourner le problème, connue sous le nom de signature basée sur l'identité, ou Identity-Based Signature (IBS). L'idée est de retourner la complexe réglementation aéronautique à notre avantage. Avant de voler, un avion doit obtenir entre autres un numéro d'immatriculation, un identifiant 24 bits de l'ICAO et un numéro de vol si c'est un avion de ligne. Toutes ces informations le caractérisent de manière unique, et sont déjà transmises à travers les messages ADS-B. N'y aurait-il donc pas une façon de s'en servir pour générer la clé publique

de l'avion ? C'est ce que propose l'IBS :

— Une autorité de certification, telle que décrite en 2.4.2.5, possède une clé secrète K_s associée à un générateur de clés privées, le Private Key Generator (PKG).

— L'autorité utilise le PKG pour, à partir de K_s et des identifiants de l'avion décrits précédemment, générer une clé privée K_{pr} qui sera fournie au propriétaire de cet avion, Bob.

— Bob utilise K_{pr} pour signer ses messages, de la même manière qu'en 2.4.2.4.

— Alice reçoit les messages ADS-B de Bob accompagnés de leur signature. Cependant, cette fois-ci, Bob ne transmet ni clé publique ni certificats correspondants.

— Alice connaît en effet l'identité supposée de Bob, car il l'a transmise à travers ses messages ADS-B. Alice est alors capable, grâce à une fonction connue de tous, de générer la clé publique K_{pb} associée à Bob (et à sa clé privée K_{pr}) à partir de ses informations d'identification.

— Alice peut donc vérifier la signature du message, toujours de la même manière qu'en 2.4.2.4.

— Il n'y a pas eu besoin de certificat. En effet, un certificat sert à faire le lien entre une clé publique et un avion donné. Comme la clé est ici générée directement à partir de l'identité de l'avion, cela n'est pas nécessaire.

Yang et al. (2014) proposent une variante, pour diminuer encore la quantité de données émises. Il s'agit de l'utilisation d'une signature qui permet la "récupération" du message. Cela signifie que seule la signature sera diffusée, et le contenu du message pourra être extrait de cette signature. Cependant, cela n'est encore une fois pas applicable pour des raisons de rétrocompatibilité. En effet, les anciens transpondeurs n'implémentant pas l'IBS ne sont pas capables d'extraire les informations de la signature.

Enfin, Cook (2015) propose d'utiliser une variante de l'IBS. Le principe de la génération des clés privées et publiques reste le même. En revanche, dans ce cas-ci, la cryptographie asymétrique va servir à un échange de clés symétriques entre Alice et Bob. Alice produira ensuite des MAC comme en 2.4.2.2 pour que Bob puisse authentifier ses messages. Il est possible d'utiliser une clé symétrique différente pour chacun des avions à proximité. Cependant, si la solution reste alors sécurisée, le fonctionnement *broadcast* de l'ADS-B est abandonné puisque des MAC différents doivent être envoyés pour chaque avion. L'autre solution consiste à utiliser la même clé avec tous les avions ; mais on retombe alors sur le problème de 2.4.2.3, où n'importe qui possédant la clé d'Alice peut se faire passer pour elle.

La solution de la cryptographie basée sur l'identité n'est donc pas directement applicable à l'ADS-B de la façon décrite dans les différents articles publiés. En effet, elle peut impliquer la transmission d'une signature pour l'authentification des messages, ce qui nécessite une trop grande utilisation de la bande passante. Les deux variantes présentées apportent aussi différents problèmes empêchant leur utilisation. L'idée est cependant judicieuse, et sera reprise au chapitre 4. Nous présenterons également en conclusion d'autres pistes d'intégration de l'IBS au sein de SAT qui pourraient être développées lors de recherches futures.

CHAPITRE 3 LE PROTOCOLE TESLA

La problématique de la diffusion en continu de données (*broadcast streaming*) est apparue au début des années 2000, avec le développement de l'Internet. Alors que les services de radio et de télévision par Internet ou satellite se démocratisaient, il fallait une solution pour sécuriser les communications. Celle-ci se devait notamment peu gourmande en bande passante, en capacité de calcul, et résistante aux pertes de paquets (un paquet étant composé d'un message et de certaines informations supplémentaires). Comme nous l'avons présenté au chapitre 2, la sécurisation de l'ADS-B doit justement répondre à ces contraintes spécifiques, qui ne sont que partiellement remplies par les solutions antérieures.

Certaines solutions avaient déjà été proposées pour répondre à la problématique de la diffusion en continu de données. Par exemple, Gennaro et Rohatgi (1997) suggéraient de signer de manière asymétrique un premier message, puis d'ensuite transmettre dans chaque paquet le hash du suivant. Cependant, le gros désavantage de cette méthode était que si un paquet était perdu, il n'était plus possible d'authentifier les messages suivants. De plus, toute la chaîne de messages devait être connue à l'avance (le hash du message n dépend du message $n+1$, qui lui-même dépend du message $n+2$ ...). Une autre solution consistant à utiliser les arbres de Merkle (Merkle, 1989) avait été proposée par Wong et Lam (1998). Là encore, il est nécessaire de connaître tous les messages initialement. Un arbre de Merkle est alors construit avec les hashs de chacun d'entre eux, et seul le hash racine est signé de façon asymétrique (figure 3.1). Afin d'être capable de remonter à la racine de l'arbre de n hashs et d'authentifier un message, celui-ci doit transmettre $\lceil \log_2(n) \rceil$ hashs, ce qui peut rapidement utiliser une grande quantité de bande passante. Plusieurs autres méthodes avaient également été suggérées, mais aucune n'était réellement satisfaisante pour répondre aux contraintes de bande passante, de capacités de calcul, et de résistante aux pertes de paquets.

Strohmeier et al. (2015a) décrivent une dernière possibilité, que nous n'avons pas encore évoquée : il s'agit de la publication rétroactive des clés. C'est une variation de la cryptographie symétrique traditionnelle, où la même clé est utilisée pour générer le code d'authentification et pour le vérifier. Cependant, au moment où le MAC est généré, la clé n'est connue que par l'émetteur. Elle est ensuite diffusée aux récepteurs à un moment ultérieur. Le MAC peut alors être vérifié et le message authentifié de manière différée. L'émetteur choisit ensuite une nouvelle clé connue de lui seul pour produire de nouveaux MAC, et le cycle recommence. C'est le fonctionnement du protocole TESLA, né d'une rencontre entre des chercheurs de l'UC Berkeley et du centre de recherche Watson, et de leur collaboration dans les années

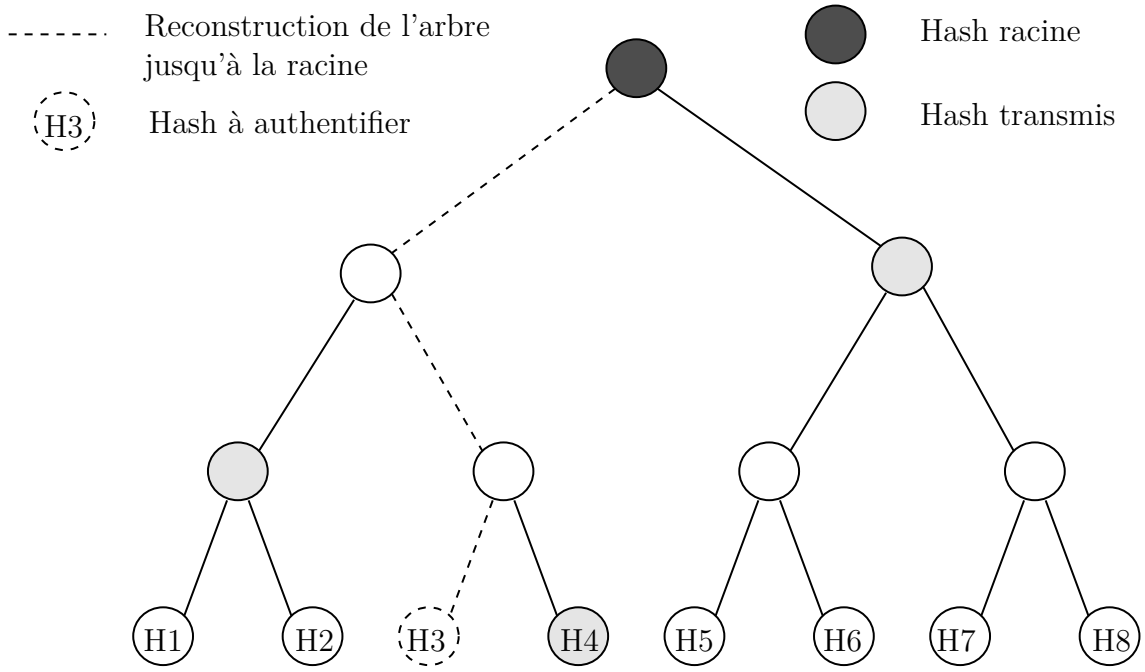


Figure 3.1 Arbre de Merkle pour les hashs des messages. Pour authentifier le hash du message 3, on doit transmettre les hashs grisés de l'arbre afin de pouvoir remonter à la racine. Celle-ci a été authentifiée au préalable de manière asymétrique.

1999 et 2000 (Perrig et al., 2000). TESLA a depuis été décrit à nouveau dans la publication *CryptoBytes* (Perrig et al., 2002a) et détaillé au sein d'une RFC pour une utilisation dans le cadre de l'internet (Perrig et al., 2005).

3.1 Solution originale

Comme nous l'avons remarqué au chapitre 2, sécuriser des communications de type *broadcast* requiert une source d'asymétrie. En effet, on veut permettre à une entité de vérifier un message sans pour autant qu'elle ne soit capable de s'authentifier elle-même comme auteur du message. Cependant, puisque les méthodes cryptographiques classiques basées sur l'asymétrie des clés impliquent une trop grande utilisation de la bande passante et des capacités de calcul, une autre forme d'asymétrie devait être trouvée. C'est ce que propose TESLA, qui se sert du temps comme source d'asymétrie (figure 3.2).

3.1.1 Préparatifs de l'émetteur

Alice, l'émettrice, divise le temps en des intervalles i de durée constante T_{int}. Elle choisit l'instant auquel démarre le premier intervalle (numéroté 0), T_0. L'intervalle i commence donc

au temps $T_i = T_0 + i * T_{int}$. Alice définit également la durée maximale T_{max} pendant laquelle elle aura à transmettre des messages. Pour la suite, nous noterons T_N le dernier intervalle. Le temps sera donc divisé en $N + 1$ intervalles, avec :

$$N + 1 \ = \ \left\lceil \frac{T_{max}}{T_{int}} \right\rceil \tag{3.1}$$

Une fonction à sens unique F est associée au protocole TESLA. Cela signifie que, connaissant une image $F(x)$ de F, il n'est pas possible de remonter à x. F se comporte en fait comme une fonction pseudo-aléatoire (PRF, Pseudo Random Function) : on ne peut pas facilement distinguer l'ensemble des images de F de celui d'une fonction aléatoire. F est connue de tous, et il peut par exemple s'agir d'une fonction de hachage cryptographique sans faille découverte.

Alice choisit alors un nombre aléatoire K_N, qui sera défini comme la clé associée à l'intervalle N. Pour chacun des autres intervalles, une clé associée sera également calculée :

$$\begin{aligned} K_{N-1} &= F(K_N) \\ &\vdots \\ K_i &= F(K_{i+1}) \\ &\vdots \\ K_0 &= F(K_1) \end{aligned} \tag{3.2}$$

Cela signifie que $K_i = F^{N-i}(K_N)$, et donc que la clé associée à chaque intervalle peut être dérivée de K_N sans devoir connaître toutes les valeurs intermédiaires.

Nous avons désormais à disposition une chaîne de clés *à sens unique* $K_0, K_1, \ldots, K_N$, chaque clé K_i étant associée à l'intervalle i. Nous choisissons alors une seconde fonction à sens unique, F', que nous appliquons à chacune des clés. Nous obtenons un nouvel ensemble de clés $K'_0, K'_1, \ldots, K'_N$. Notons qu'il ne s'agit plus d'une chaîne de clés ; la connaissance de K'_{i+1} ne permet pas de déduire K'_i. Cet ensemble de clés nous servira à authentifier les messages. Nous aurions pu directement utiliser la chaîne de clés K, cependant pour des raisons de sécurité il est généralement recommandé d'utiliser des clés distinctes pour des opérations distinctes. Une clé K_i étant déjà utilisé pour calculer K_{i-1}, il est donc préférable de dériver une clé K'_i de K_i pour les opérations d'authentification. Pour cela, une seconde fonction à sens unique F' est choisie, et $K'_i = F'(K_i)$.

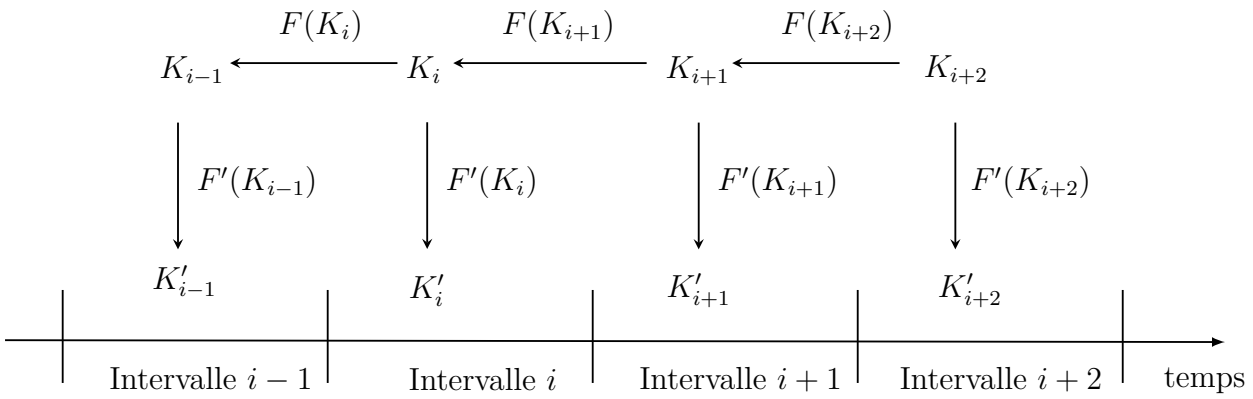


Figure 3.2 La chaîne de clés TESLA est générée à partir de K_N, en lui appliquant une fonction à sens unique F. Le temps est divisé en intervalles, et à chacun d'entre eux est associé une clé dérivée de cette chaîne grâce à une autre fonction à sens unique F'.

3.1.2 Émission d'un message et distribution des clés

Soit t_e le moment où Alice veut émettre un message. t_e se trouve dans l'intervalle i, c'est-à-dire que $T_i < t_e < T_{i+1}$. Alice va donc utiliser la clé K'_i associée à cet intervalle, qui ne sera alors connue que d'elle seule, pour calculer un MAC qui servira à authentifier son message.

L'horloge d'Alice et celles de tous les récepteurs, dont Bob, sont synchronisées. Cependant, il ne s'agit pas d'une synchronisation absolue. En réalité, Bob, dont l'horloge indique un temps t_B, n'a besoin de savoir que l'écart maximal qui peut le séparer de l'horloge t_A d'Alice. Si $t_A = t_B + \delta$, alors Bob n'a besoin que de connaître Δ tel que $\delta < \Delta$. On nomme cela *loose time synchronization*.

Pour que Bob soit capable de vérifier le MAC du message d'Alice, il faut qu'il connaisse la clé qui a été utilisée pour le générer. Cependant, lors de l'intervalle i pendant lequel K'_i est utilisée par Alice, la clé ne doit être connue que d'elle seule. Alice introduit donc un délai de d intervalles pour la diffusion des clés. La clé K_i dont est dérivée K'_i ne sera donc diffusée qu'à l'intervalle $i + d$. d doit être choisit de telle sorte que Bob soit sûr que la clé n'ait pas encore été diffusée à l'instant où il reçoit le message d'Alice :

$$T_{trans} + \Delta < d * T_{int} \qquad \Leftrightarrow \qquad d > \frac{T_{trans} + \Delta}{T_{int}} \tag{3.3}$$

Dans le cas où il y a plusieurs récepteurs, Alice doit choisir d de façon à ce que l'inégalité soit respectée pour chacun d'entre eux. Dans le cas contraire, le message ne pourra pas être

authentifié, car Oscar, un attaquant, pourrait déjà être en possession de la clé et avoir forgé celui-ci.

En fin de compte, un message M_j émis par Alice au cours de l'intervalle i sera transmis au sein d'un paquet P_j de la façon suivante :

$$P_j = \{M_j \parallel MAC(K'_i, M_j) \parallel K_{i-d}\} \tag{3.4}$$

3.1.3 Authentification des messages

Pour être capable d'authentifier les messages d'Alice, Bob doit tout d'abord connaître certaines informations :

- Les fonctions à sens unique F et F' utilisées
- L'heure de début du premier intervalle T_0 et la durée d'un intervalle T_{int}
- Le délai de publication des clés d choisi par Alice
- La dernière clé de la chaîne, K_0 ou toute autre clé appartenant à la chaîne et déjà publiée

Celles-ci devront avoir été reçues de manière authentifiée en amont de la communication, en utilisant par exemple de la cryptographie asymétrique classique et une ICP.

À la réception d'un paquet P_j, Bob procède de la manière suivante :

1. Bob connaît déjà au moins une clé K_v déjà vérifiée, c'est à dire dont il est certain qu'elle appartient à la chaîne d'Alice. Il peut s'agir de K_0 ou d'une clé plus récente et déjà publiée. Il applique F à K_{i-d} un nombre de fois θ, tel que $K_v = F^\theta(K_{i-d})$. On obtient donc :

$$(i-d) - v = \theta \qquad \Leftrightarrow \qquad i = \theta + d + v \tag{3.5}$$

   Bob sait alors au cours de quel intervalle i le message a été envoyé. De plus, puisque K_v était vérifiée, et que appliquer F^θ à K_{i-d} permet de retomber sur K_v, c'est que K_{i-d} appartient bien à la chaîne de clés d'Alice et cette nouvelle clé est donc authentifiée.

2. Bob vérifie que l'instant t_r où il a reçu le paquet vérifie bien $t_r + \Delta < T_{i+d}$. Si c'est le cas, il est assuré que Alice n'a pas encore diffusé sa clé, et donc que la signature du paquet est valide. Il ajoute alors le triplet $\Big(i, M_j, MAC(K'_i, M_j)\Big)$ dans une file d'attente, que

nous nommerons également *tampon* par la suite. Dans le cas contraire, la clé peut avoir été diffusée et Bob supprime donc le paquet.

3. Bob continue de recevoir des paquets et de les enregistrer dans son tampon le cas échéant, jusqu'au moment où il reçoit K_i. Comme à l'étape 1, il vérifie alors qu'elle appartient bien à la chaîne d'Alice. Si la clé est authentifiée, Bob applique F' à K_i et obtient K'_i. Il peut alors vérifier le MAC de M_j, et donc s'assurer que le message provenait bien d'Alice. Si d'autres messages se trouvent dans le tampon, avec des MAC générés pas K'_i ou des clés plus anciennes, il est aussi capable de les authentifier.

3.2 Déplacement de la file d'attente vers l'émetteur

Peu après avoir décrit la première version de leur protocole TESLA, Perrig et al. (2001) ont fait le constat que la mise en mémoire tampon des paquets au niveau du récepteur peut causer plusieurs problèmes. Tout d'abord, il faut être capable de stocker tous les messages dont la clé n'a pas encore été reçue. Si d est grand et que l'on est en présence d'un grand nombre d'émetteurs, cela peut vite constituer une quantité importante de données. De plus, cela rend vulnérable à une attaque de type déni de service (DoS, Denial of Service). Cela signifie qu'un attaquant pourrait volontairement saturer l'espace disponible dans la mémoire tampon en envoyant une quantité importante de paquets et ainsi empêcher les paquets légitimes d'être stockés.

La solution apportée est de déplacer la file d'attente du récepteur vers l'émetteur. Ce dernier va mettre de côté tous les messages générés pendant un temps $T_{int} * d$, soit le délai d'authentification pour la version originale de TESLA. Ainsi, si v messages sont générés par intervalle, Alice va attendre pour émettre M_j que M_{j+vd} soit connu. Elle pourra ainsi calculer un hash de M_{j+vd}, $H(M_{j+vd})$. On pose :

$$D_j = \Big\{ M_j \parallel H(M_{j+vd}) \Big\} \tag{3.6}$$

Alice émettra alors :

$$P_j = \Big\{ D_j \parallel MAC(K'_i, D_j) \parallel K_{i-d} \Big\} \tag{3.7}$$

Au moment où il recevra M_{j+vd}, Bob aura déjà reçu $H(M_{j+vd})$ dans le message M. M_{j+vd} sera donc directement authentifié, sans avoir besoin de l'enregistrer dans un tampon. Dans le cas où M aurait été perdu, M_{j+vd} peut cependant toujours être authentifié de façon traditionnelle.

Si l'authentification est ici immédiate, la diffusion des messages est quant à elle retardée. Or,

dans le cas de l'ADS-B, on veut que les données de position soient connues en temps réel. On ne peut donc pas tolérer un retardement de l'envoi des messages au niveau de l'émetteur, alors que l'on peut accepter un certain délai pour l'authentification comme expliqué au chapitre 2. Cette optimisation n'est donc pas convenable.

De plus, la menace du DoS sur un récepteur n'est en réalité pas très forte. En effet, les messages ne se trouvent dans le tampon que pour un temps réduit de $T_{int} * d$. La bande passante étant limitée, un envoi massif de messages provoquerait probablement un déni de service sur le canal de transmission plus rapidement que sur le tampon, si ce dernier est d'une taille raisonnable (quelques Mo).

3.3 μTESLA pour les réseaux de capteurs

Le protocole TESLA a largement été repris dans le domaine des réseaux de capteurs, puisqu'il répond à la problématique de l'authentification de messages *broadcast* avec peu de bande passante disponible. Or l'architecture de ces réseaux de capteurs est très semblable à celle de l'ATC. Elle est constituée de *nœuds*, à savoir les capteurs, qui diffusent de manière régulière leurs mesures à un petit nombre de *stations de base*, qui agrègent ces données et les transmettent sur un réseau extérieur. On reconnaît donc bien la similitude avec les avions, qui comme les *nœuds* diffusent leurs informations périodiquement, et les antennes de l'ATC, qui agrègent ces informations et les transmettent aux écrans des contrôleurs.

Puisque l'architecture des réseaux de capteurs est semblable à celle de l'ATC et possède un certain nombre de problématiques communes, il est intéressant d'étudier les optimisations de TESLA qui ont été proposées pour ce domaine. Cependant, il est important de prendre en compte deux points différenciateurs lors de cette étude, qui pourraient empêcher la transposition des solutions apportées. Tout d'abord, dans le cas des réseaux de capteurs, la station de base peut servir d'intermédiaire pour les échanges entre nœuds. Dans le cas de l'ADS-B, l'objectif est au contraire que les avions puissent s'échanger leur position sans l'intervention des contrôleurs. De plus, chaque capteur a peu de capacité de stockage et des capacités de calcul très limitées, d'autant plus qu'il doit pouvoir fonctionner plusieurs années alimenté par une petite batterie. Les capacités de stockage et de calcul d'un transpondeur sont certes limitées, mais dans une bien moindre mesure. Il n'a notamment pas de problème d'économie d'énergie.

Perrig et al. (2002b) décrivent une évolution du protocole TESLA pour les réseaux de capteurs nommée μTESLA. L'optimisation proposée consiste à remplacer la signature électronique servant à l'authentification initiale de la chaîne de clés de l'émetteur. En effet, celle-ci utilise

de la cryptographie asymétrique, qui demande de trop grosses ressources de calcul pour les capteurs. La solution est alors que chaque nœud partage une clé secrète qui lui est propre avec la station de base. Lorsqu'un nœud A voudra commencer à authentifier les paquets d'un autre nœud B, il contactera la station de base. Celle-ci, n'ayant pas de contrainte de capacité de stockage, connaît les paramètres TESLA de tous les nœuds. Elle pourra donc transmettre les paramètres du nœud B au nœud A, de façon sécurisée grâce à un MAC (cryptographie symétrique) généré à partir de la clé symétrique qu'elle partage avec le nœud A. De plus, puisqu'un nœud n'a pas les capacités de stockage pour enregistrer toute sa chaîne de clés, la station de base va effectuer le stockage à sa place. Lorsqu'un nœud aura besoin d'émettre et donc d'obtenir ses clés, il en fera la demande à la station de base. Celle-ci lui transmettra de façon chiffrée grâce à la clé qu'ils partagent.

Dans le cas de l'ATC, les transpondeurs ont les capacités de calcul suffisantes pour utiliser de la cryptographie asymétrique. Cependant, les clés et signatures de cette dernière sont longues, et il pourrait être intéressant de s'en passer pour économiser de la bande passante. Mais la solution proposée implique que l'ATC serve d'intermédiaire pour l'authentification de chaque avion, ce qui n'est pas souhaitable. On se retrouve en réalité avec une solution très semblable à celle proposée par Viggiano et al. (2010) et présentée en 2.4.2.3, avec donc les mêmes limitations.

Liu et Ning (2004) proposent une autre optimisation intéressante permettant d'augmenter la durée de vie d'une chaîne de clés, tout en conservant un temps T_{int} relativement court (ce qui permet de réduire le délai d'authentification), et en n'ayant qu'un nombre réduit de clés à stocker. L'idée est d'utiliser plusieurs sous-chaînes C_i avec ce court T_{int}, qui seront elles-mêmes authentifiées par une autre chaîne *maître* C_M, avec un T_{int} beaucoup plus long. La chaîne maître C_M durera donc toute la durée de l'échange, sans toutefois qu'elle ne contienne trop de clés (figure 3.3)

Comme toute chaîne TESLA, C_M est divisée en intervalles. À l'intervalle i de C_M est attachée la chaîne C_i qui servira à authentifier les messages pendant toute la durée de cet intervalle. La sous-chaîne C_i est divisée en N intervalles, et $K_{i,N}$ est choisie aléatoirement. Chaque intervalle j de la sous-chaîne possède une clé $K_{i,j}$ dérivée de $K_{i,N}$ grâce à une fonction de hachage connue.

Pour pouvoir authentifier les messages émis pendant l'intervalle i, il faut donc connaître et authentifier la clé $K_{i,0}$ auparavant. C'est ce à quoi va servir la chaîne maître C_M. Supposons que l'on ait choisi un délai d'un intervalle ($d = 1$) pour C_M, et que cette chaîne de clés maître a déjà été authentifiée de manière asymétrique. À l'intervalle $i-2$, l'émetteur va transmettre la clé $K_{i,0}$ dans un message TESLA. À l'intervalle $i-1$, la clé K_{i-2} sera dévoilée et $K_{i,0}$ sera

alors vérifiée. Les messages pourront donc être authentifiés à l'intervalle suivant i grâce à la chaîne TESLA C_i authentifiée par $K_{i,0}$.

Reste un petit problème. Une des grandes forces de TESLA est d'être tolérant à la perte de messages. Si un paquet P_i est perdu et que des messages précédents n'ont pas été authentifiés, la réception de P_{i+1} permettra de redescendre dans la chaîne de clés et ainsi de récupérer celles manquantes pour l'authentification des messages en attente. La méthode proposée ici casse ce principe en découpant la chaîne d'authentification des messages en sous-chaînes indépendantes. Par exemple, la perte de $K_{i,N}$ ne pourra pas être compensée par la réception de $K_{i+1,0}$, car il n'y a pas de relation entre les clés des chaînes C_i et C_{i+1}.

Pour résoudre cela, une solution est proposée. Plutôt que de choisir $K_{i,N}$ de manière aléatoire, on utilise une fonction de hachage H et on choisit $K_{i,N} = H(K_{i+1})$. Durant l'intervalle i, K_{i+1} n'est pas dévoilé et donc nul autre que l'émetteur ne peut connaître $K_{i,N}$. Cependant, à l'intervalle $i + 1$, K_{i+1} sera dévoilée. Cela permettra à n'importe qui de calculer $K_{i,N}$ et donc d'authentifier tous les messages émis pendant l'intervalle i.

Cette solution paraît intéressante pour les réseaux de capteurs aux capacités de stockage réduites. En effet, cela évite à la station de base de servir d'intermédiaire pour la transmission des clés à l'émetteur. Cependant, dans le cas des transpondeurs, la mémoire permet d'enregistrer un nombre important de clés. De plus, comme nous allons l'expliquer en 4.2.1, les avions ont plusieurs possibilités pour mettre à jour leur chaîne de clés de façon régulière. Nous n'aurons donc pas à stocker de trop longues chaînes. En conséquence, nous ne mettrons pas en application cette optimisation.

Enfin, Liu et al. (2005) proposent une nouvelle méthode pour l'échange des paramètres TESLA initiaux, qui ne requiert ni cryptographie asymétrique ni d'utiliser la station de base comme intermédiaire. Cette solution fait intervenir un arbre de Merkle. Supposons que le réseau est constitué de m nœuds et stations de base. Une autorité de certification génère avant tout échange m instances de TESLA, une pour chacun des acteurs. Notons S_j les données d'authentification (durée de l'intervalle, K_0, ...) de l'acteur j, et soit H une fonction de hachage utilisée par la CA. Cette dernière calcule pour chaque acteur j le hash $h_j = H(S_j)$ et construit un arbre de Merkle à partir de ces hashs (comme présenté à la figure 3.1, mais cette fois-ci avec les hashs des données d'authentification à la place des hashs des messages). La racine de l'arbre h_R doit ensuite être diffusée à tous les participants, mais cela peut par exemple se faire lors de la programmation initiale du capteur. Un certificat de l'instance TESLA d'un acteur j sera alors construit à partir de tous les hashs qui sont nécessaires pour, à partir de S_j, remonter à la racine h_R. Cela correspond à $\lceil \log_2(m) \rceil$ hashs à diffuser pour l'authentification d'un acteur.

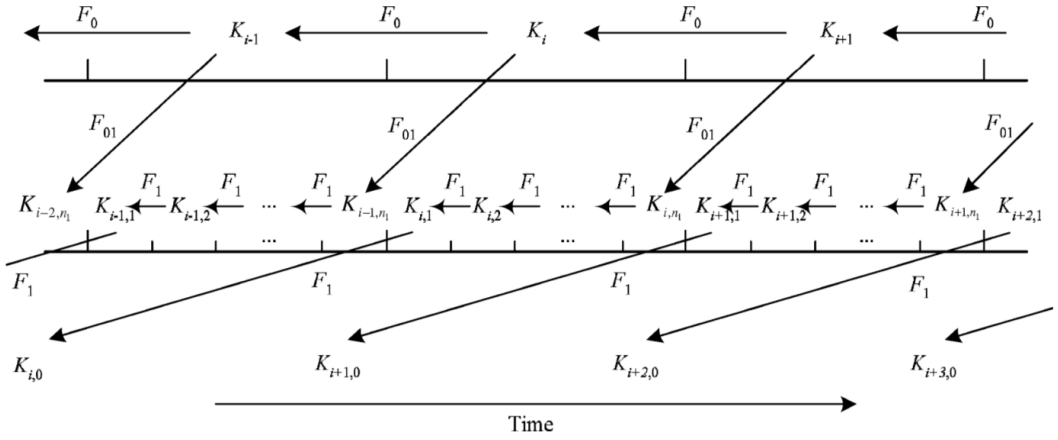


Figure 3.3 Sous-chaînes TESLA telles que proposées par Liu et Ning (2004). Une chaîne principale sert à produire les clés génératrices des sous-chaînes.

Ceci est une solution intéressante pour les réseaux de capteurs, car cela retire le délai produit avec μTESLA dû à l'utilisation de la station de base comme intermédiaire. Par comparaison à une signature numérique, celle-ci est remplacée par la transmission et le calcul de $\lceil \log_2(m) \rceil$ hashs pour l'arbre de Merkle, plus un pour le calcul de S_j. Ceci ne présente une amélioration que pour un nombre réduit de participants, qui dépend de la taille des hashs et des clés asymétriques choisies. Au niveau du trafic aérien, on compte actuellement 25 000 avions de ligne en circulation dans le monde, et ce nombre devrait plus que doubler au cours des vingt prochaines années. Il faut également ajouter à cela tous les avions privés. Puisqu'une fois généré, l'arbre de Merkle ne peut être modifié, il faut prévoir un nombre de participants suffisant pour prévoir la croissance des prochaines années. En considérant que 100 000 chaînes de clés devraient être générées (cela suppose qu'une seule chaîne TESLA sera suffisante par avion, en utilisant par exemple la méthode des sous-chaînes introduite précédemment), cela représente 17 hash à diffuser pour chaque authentification. Si on ne dépasse pas la taille d'une signature numérique asymétrique, on s'en rapproche au moins très fortement. La complexité ajoutée par cette solution par rapport à une ICP classique ne justifie alors pas son utilisation. De plus, les ICP sont plus naturels dans ce cas, comme nous l'expliquerons à la section 4.2.2.

CHAPITRE 4 SAT : TESLA ADAPTÉ POUR l'ADS-B

Utiliser TESLA pour sécuriser l'ADS-B paraît prometteur, puisque le protocole cumule les avantages de la cryptographie symétrique (des codes d'authentification courts plutôt que de longues signatures), tout en conservant une asymétrie entre l'émetteur et le récepteur (seul le premier connaît la clé au moment où celle-ci est valide pour la génération du MAC). Cependant, si cette méthode a souvent été abordée de façon succincte par de nombreux chercheurs (Cook, 2015; Haomiao et al., 2013; Strohmeier et al., 2015a), elle n'a jamais été étudiée de façon suffisamment détaillée pour déterminer son applicabilité, sa performance, et sa viabilité dans le cadre de l'ADS-B.

Nous proposons donc notre propre adaptation de TESLA pour l'ADS-B, que nous nommons SAT. Le choix des paramètres de SAT sera discuté au chapitre 5, ainsi que la comparaison de ses performances avec les solutions existantes.

4.1 Synchronisation des horloges

Comme nous l'avons vu au chapitre 3, TESLA repose largement sur le temps et demande au minimum une synchronisation relative entre l'émetteur et les récepteurs. Dans la solution originale, Perrig et al. (2002a) proposaient un protocole de synchronisation reposant sur une négociation entre l'émetteur et chaque récepteur. On pouvait alors déterminer Δ, l'écart maximum entre leurs deux horloges. À partir de Δ, on était capable de déterminer le délai d optimal avant la diffusion des clés.

Cependant, cette technique présente deux problèmes majeurs dans le cadre de l'ADS-B. Tout d'abord, elle requiert un échange bidirectionnel entre chaque récepteur et l'émetteur. Cela est contraire au principe de communication uniquement *broadcast* que l'on veut conserver. De plus, la valeur optimale de d dépend de Δ, et est propre à chaque récepteur. Il faudrait donc *a priori* utiliser une chaîne TESLA différente pour chaque récepteur. Ceci implique d'envoyer un MAC différent pour chacun d'entre eux dans chaque message, ce qui augmente largement l'utilisation de bande passante et n'est pas envisageable. On pourrait également borner Δ, et choisir un d en conséquence. Cependant, un avion dont le défaut de synchronisation avec l'émetteur dépasserait Δ ne serait pas capable d'authentifier les messages.

4.1.1 Utilisation du temps GNSS

Heureusement, les avions utilisant l'ADS-B sont équipés de systèmes de navigation par satellite, ou GNSS. En effet, c'est de ces systèmes que proviennent les informations de positionnement qu'ils transmettent. Il peut s'agir du Global Positioning System (GPS) américain, du GLObalnaya NAvigatsionnaya Sutnikovaya Sistema (GLONASS) russe, du Galileo européen ou encore du COMPASS chinois (Hegarty et Chatre, 2008).

Le positionnement GNSS repose sur la multilatération. En connaissant au minimum les positions de quatre satellites, et en utilisant les différences de temps d'arrivée (TDOA) entre les signaux émis simultanément par chacun de ces satellites, on est capable de calculer notre position. Il s'agit en fait de la situation symétrique de la multilatération présentée en 2.4.1.1, où l'on souhaitait calculer la position d'un avion à partir de son signal reçu par quatre antennes fixes.

Puisqu'il faut connaître la position des satellites pour être capable d'effectuer la multilatération, le signal envoyé par chaque satellite contient des informations sur son orbite. À partir de celles-ci, la position du satellite peut être calculée. Pour être capables d'émettre un signal de façon simultanée, les satellites embarquent une horloge atomique, très précise (une déviation de moins de 10^{-16} seconde chaque seconde). De plus, une horloge maître synchronise régulièrement les horloges des différents satellites entre eux, pour encore plus de précision. Le temps exact d'émission du signal est également encodé dans celui-ci.

Une fois que l'avion a déterminé sa position grâce à la multilatération, il est capable de savoir la distance qui le sépare d'un satellite donné. Il peut alors calculer le temps de propagation du signal, et en déduire le temps actuel. Celui-ci est utilisé pour resynchroniser l'horloge de l'avion avec le temps GNSS exact. En effet, une horloge atomique coûte très cher, et les avions ne sont donc équipés que d'horloges au quartz. Ces dernières ne restent suffisamment précises pour les calculs de positionnement que pendant quelques secondes.

Selon la spécification du GPS, la précision alors obtenue pour l'horloge de l'avion par rapport à l'horloge maître sur laquelle se synchronisent les satellites est de l'ordre de 40 ns, et d'au pire 100 ns (GPS directorate, 2013). Cela est déjà très précis, mais des techniques de correction du positionnement ont été introduites pour encore améliorer ces résultats. En effet, l'orbite du satellite est sujette à de légères variations. Des corrections sont donc diffusées à travers des éphémérides. De plus, les horloges sont sujettes à de faibles dérives constantes qui peuvent être prédites. Enfin, des éruptions solaires peuvent produire des perturbations ionosphériques qui introduisent un délai dans la propagation du signal des satellites. Chaque système GNSS implémente donc son propre système de corrections ; dans le cas du GPS, il s'agit du Wide

Area Augmentation System (WAAS) géré par la FAA.

En conclusion, même en considérant une erreur maximale de synchronisation de 100 ns avec l'horloge maître, on obtient $\Delta = 200\ ns$ (dans le pire des cas, l'émetteur a 100 ns d'avance sur l'horloge maître et le récepteur est 100 ns en retard). Cela ne nécessite de négociation entre l'émetteur et le récepteur, et cette borne supérieure est commune à tous les avions. Une seule chaîne de clés TESLA et donc un unique MAC sera nécessaire pour la diffusion de chaque message.

4.1.2 Sécurité du temps GNSS

Étant donné que TESLA repose sur le temps pour vérifier l'authenticité d'un message, il faut s'assurer de la fiabilité de l'horloge. Comme nous venons de le voir, la précision de celle-ci est *a priori* assurée par le GNSS. Cependant, comme tout système, celui-ci peut être la cible d'une attaque. Ainsi, il pourrait être possible de manipuler le signal GNSS afin de retarder l'horloge d'un récepteur ou d'avancer l'horloge de l'émetteur. L'erreur de synchronisation pourrait alors dépasser Δ, et la clé d'un intervalle être diffusée alors qu'elle paraît encore valide pour l'authentification de cet intervalle.

Cependant, le signal GNSS est déjà utilisé pour l'obtention du positionnement d'un avion qui est diffusé par ADS-B. Même en se servant d'une autre source d'horloge sécurisée pour l'authentification des messages, leur contenu pourrait de toute façon être erroné bien que leur intégrité soit assurée (puisque l'erreur a été introduite à la source et non pendant la transmission).

La sécurité du GNSS est loin d'être une problématique nouvelle. En effet, c'est une technologie qui a été introduite en premier lieu par l'armée américaine dans les années 1980, et de nombreux travaux ont été consacrés à ce sujet (Volpe, 2001; Humphreys et al., 2008; Papadimitratos et Jovanovic, 2008). L'étude et la résolution de ce problème sont cependant hors de la portée de ce mémoire. Il est toutefois intéressant de mentionner les travaux de Becker et al. (2009) et de Lo et Enge (2010), qui proposent justement l'utilisation de TESLA pour sécuriser le GNSS.

4.2 Authentification de la chaîne de clés d'un avion

Comme expliqué au chapitre 3, l'utilisation de TESLA pour authentifier un message requiert que la chaîne de clés de l'émetteur soit elle-même authentifiée. Puisque les différentes alternatives décrites pour les réseaux de capteurs ne sont pas adaptées à nos propres contraintes, nous proposons d'utiliser de la cryptographie asymétrique classique. Son utilisation pour la

signature des messages, envoyés plusieurs fois par secondes, était bien trop coûteuse en bande passante. C'est pourquoi nous avons choisi d'utiliser TESLA. En revanche, la diffusion des certificats pour l'authentification de la chaîne de clés est beaucoup moins fréquente, puisque ceux-ci ne doivent être obtenus qu'une seule fois par un nouveau récepteur. L'utilisation de certificats électroniques est donc acceptable en termes de performance.

4.2.1 Infrastructure à clé publique

Tout processus de certification requiert une infrastructure à clé publique (ICP), c'est-à-dire une hiérarchie d'autorités de certification (CA) comme décrite en 2.4.2.5. Cette ICP possède à sa base une ou plusieurs autorités racines, que l'on considère comme dignes de confiance. Ces autorités racines ne produisent généralement pas directement de signatures pour des messages, mais s'occupent plutôt de certifier des sous-autorités. Ces dernières pourront soit à leur tour certifier de nouvelles sous-autorités, ou produire des signatures de messages.

Dans le cas du contrôle du trafic aérien, chaque avion est immatriculé dans un certain pays, auprès de l'autorité locale responsable de l'aviation civile (NAA). Ce processus d'immatriculation est en réalité une certification de l'avion, l'autorisant à voler. Par exemple, aux États-Unis, la FAA demande un certificat d'achat de l'avion et vérifie l'identité du propriétaire grâce à un formulaire présenté en Annexe E. Chaque avion immatriculé obtient alors un code, dont le format est défini par l'OACI.

Le fait qu'une instance internationale, l'OACI, supervise le processus d'immatriculation joue un rôle important dans le processus de contrôle du trafic aérien. Tout d'abord, un avion immatriculé dans un pays donné doit pouvoir être identifié à l'extérieur de celui-ci. Les bases de données d'immatriculation doivent donc être partagées. De plus, on ne veut pas que deux avions enregistrés dans deux pays distincts puissent se retrouver avec le même numéro d'immatriculation, par manque de coordination.

Cela joue un rôle encore plus important dans notre contexte de sécurité. Nous pourrions choisir de définir les NAA de chaque pays comme autorités de certification racine. Cela serait techniquement possible, puisqu'il suffirait d'enregistrer les clés publiques des 197 pays reconnus par l'ONU dans les transpondeurs de chaque avion. Cependant, parmi ces 197 pays, certains sont politiquement instables ou peuvent avoir des gouvernements fortement corrompus, et d'autres manquent de moyens pour financer une autorité de l'aviation civile. Certains de ces 197 pays n'appartiennent même pas à l'OACI, qui ne compte que 191 membres. Il serait donc très facile pour un attaquant d'aller dans un pays un peu moins minutieux dans ses vérifications pour obtenir, possiblement en échange de quelques billets, des immatriculations d'avions inexistants. Chaque pays devrait donc en permanence évaluer la fiabilité des autres

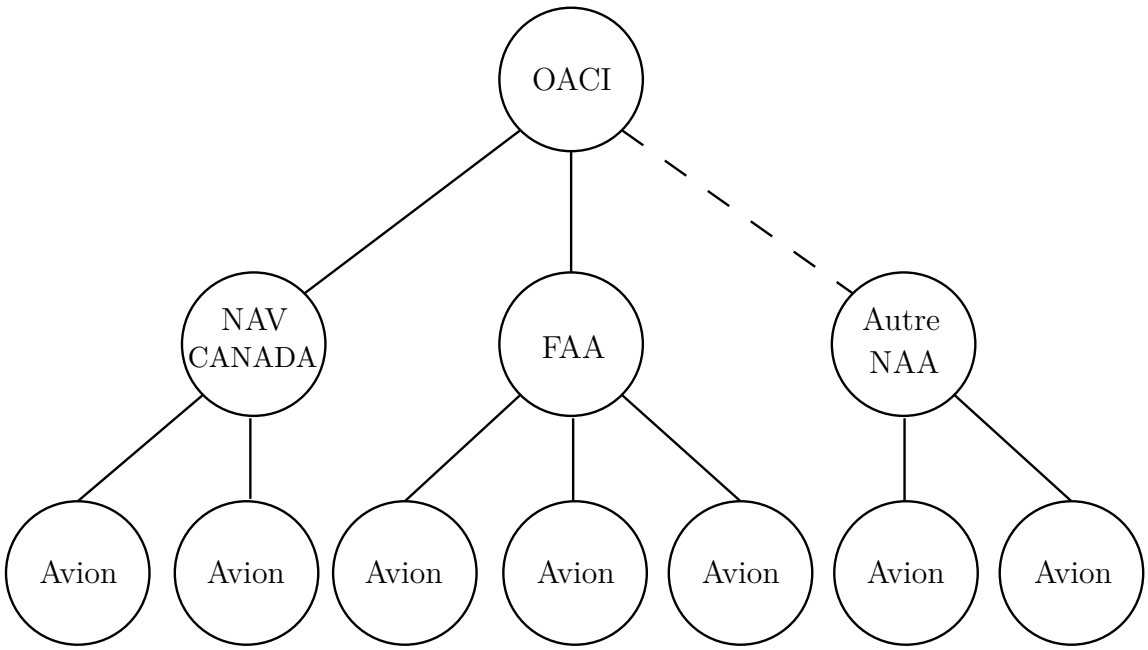


Figure 4.1 Infrastructure à clés publiques pour l'authentification de la chaîne de clés d'un avion. L'OACI certifie les NAA de chacun de ses pays membres. Les NAA s'occupent quant à elles de produire les certificats pour les avions dont elles ont émis l'immatriculation.

nations, et possiblement en retirer certaines de la liste des autorités de certification racine.

C'est pourquoi nous choisissons de nommer l'OACI comme unique autorité racine chargée de la certification des avions. Chaque NAA lui délègue déjà une partie de ses pouvoirs en termes de réglementation de l'aviation civile, et des inspecteurs sont chargés de vérifier que toutes les normes définies par l'organisation sont bien respectées par chaque pays. Le principe sera exactement le même pour la certification SAT. Une série de règles sera définie pour s'assurer que seul un avion réel peut obtenir une immatriculation, et des inspecteurs seront dépêchés pour en vérifier l'application. Les NAA qui respectent tous les critères obtiendront alors des certificats qui feront d'elles de sous-autorités de certification. Ces certificats pourront être révoqués si des anomalies sont découvertes.

On peut se demander si l'OACI possède l'expertise nécessaire pour devenir une autorité de certification racine. En effet, cela demande une grande expérience pour éviter toute perte de clé et pour s'assurer que les NAA respectent bien toutes les règles requises. Cependant, l'OACI est en réalité déjà une autorité de certification racine pour l'émission des passeports électroniques (OACI, 2011). Elle s'occupe donc déjà de gérer le répertoire à clés publiques (PKD, Public Key Directory) des sous-autorités certifiées pour délivrer de tels passeports. Ce répertoire pourrait donc facilement être étendu pour y inclure les clés publiques des NAA

autorisées pour la certification des chaînes de clés SAT.

4.2.2 Processus de certification de l'avion

Chaque avion est donc désormais en possession de la liste des clés publiques des sous-autorités de certification autorisées par l'OACI. Dans la version traditionnelle de TESLA, la CA signe directement la chaîne de clés de l'émetteur, qu'il conserve pour la totalité de sa durée de vie. Un avion qui aurait besoin de s'authentifier diffuserait donc simplement sa clé K_0 et le certificat correspondant.

4.2.2.1 Certification de l'avion par la NAA

Cependant, dans notre contexte, cela pose plusieurs problèmes. Tout d'abord, un avion a généralement une durée de vie de plusieurs dizaines d'années. Il faudrait donc une très longue chaîne de clés. La méthode de chaînes à plusieurs niveaux présentée en 3.3 pourrait être utilisée. Cependant, utiliser une même chaîne pendant plusieurs années et avec de longs intervalles (pour la chaîne principale) augmente considérablement la probabilité de trouver une collision sur la fonction de hachage utilisée pour produire la chaîne, et ainsi de découvrir des clés futures. De plus, chaque nouveau récepteur doit redescendre toute la chaîne jusqu'à K_0 afin d'être capable de l'authentifier, ce qui peut introduire un délai pour la première authentification (si la chaîne comporte plusieurs milliers de clés, cela implique autant de milliers de hashs à calculer). Enfin, cela implique de connaître l'instant exact où a débuté la chaîne, pour être capable de connaître le numéro de l'intervalle actuel.

Nous proposons donc une autre solution. Elle consiste à accorder à un avion le rôle de sous-autorité de certification pour ses propres chaînes de clés. Chaque avion obtiendra son certificat de CA de la part de la NAA de son pays d'attache. Tous les avions produiront alors de nouvelles chaînes de manière périodique, ce qui diminuera leur taille. De plus, le premier intervalle des chaînes de tous les avions commencera au même instant. Cela retirera la nécessité de diffuser ce paramètre et facilitera la synchronisation, puisque tous les avions se trouveront toujours dans le même intervalle (aux erreurs d'horloge près).

Nous proposons de renouveler la chaîne de clés pour chaque vol. En effet, un renouvellement plus court (en cours de vol), casserait la possibilité d'authentifier les paquets d'une chaîne antérieure du vol, si les clés correspondantes n'ont pas été reçues. Il n'est en revanche pas utile de pouvoir authentifier les messages émis lors d'un ancien vol. Nous choisissons donc de produire des chaînes d'une durée de 48 heures, débutant à minuit le jour du décollage. L'autonomie maximum de vol actuelle pour les avions les plus performants étant d'une vingtaine

d'heures, cela laisse une marge suffisante. Nous détaillerons ce choix en 5.7.

Reste cependant le problème de la production du certificat d'un avion. Comme expliqué en 4.2.1, une première possibilité consiste à fournir celui-ci à un avion en même temps que son immatriculation. Nous nommons cela un certificat à *long terme*. Cependant, que se passe-t-il si la clé de l'avion est volée, ou si l'avion est mis hors service ? Son certificat devrait être révoqué. Cela implique l'utilisation de listes de révocations (CRL, Certificate Revocation List). Une autre possibilité consiste à donner une date d'expiration aux certificats, par exemple au bout d'un an. Les clés de l'avion et son certificat seraient alors renouvelés une fois par an lors d'une maintenance. Mais cela ne résout pas le problème d'une clé volée en cours d'année.

L'autre possibilité consiste à procurer un nouveau certificat à un avion avant chaque vol, dont la durée de validité ne serait que de 48 heures (la durée de la chaîne produite pour le vol). Cette période très courte, où l'avion est en vol, limite très fortement le risque et les conséquences d'un vol de clé. Cependant, celle-ci nécessite la réception du certificat avant chaque décollage, et requiert donc un canal de transmission sécurisé servant à authentifier l'avion. De plus, la NAA devra déléguer son autorité de certification à chacun des aéroports de son pays, qui seront en charge de s'assurer que les clés publiques qu'ils certifient appartiennent bien à des avions légitimes.

Nous discutons ci-après les critères d'utilisation de chacun de ces deux types de certificats, à *long* et *court* terme, pour savoir lequel d'entre eux utiliser selon le type de vol, et les conséquences qui en découlent.

4.2.2.2 Certificat à long terme

Un avion devra toujours posséder un certificat à long terme qui, comme expliqué précédemment, lui sera remis par la FAA lors de son immatriculation, puis renouvelé au minimum annuellement au cours d'une mise à jour de l'avionique en période de maintenance. Ce sera ce certificat qui sera utilisé par défaut. En effet, dans de nombreux pays en développement ou dans de petits aéroports, le seul canal de communication est la radio. Cela rend difficile le processus d'authentification d'un avion par la NAA et la transmission de sa clé publique et du certificat associé.

Le niveau de sécurité est bien sûr inférieur à un certificat à *court* terme, puisque des clés pourraient être volées pendant la période de validité du certificat. Cela limite toutefois déjà largement les capacités d'un attaquant à produire une multitude d'avions fantômes, puisqu'il faudrait qu'il ait réussi à se procurer autant de clés. Un vol de clés à grande échelle serait de plus probablement vite découvert.

Une liste de révocation (CRL) sera mise en place, qui comprendra toutes les clés ayant été volées, ayant servi à authentifier de faux messages, ou encore appartenant à des transpondeurs hors service. Cela permettra d'ignorer les messages signés grâce à de tels certificats. La CRL sert gérée de façon centralisée par l'OACI, pour faciliter sa mise à jour. Chaque NAA se chargera d'y ajouter les clés sous son contrôle qui semblent frauduleuses. La mise à jour de la CRL ne nécessitera ainsi le téléchargement que d'une seule liste plutôt que d'une par pays.

Cependant, même avec une seule CRL centralisée, mettre à jour le transpondeur d'un avion afin d'y intégrer la dernière version de cette CRL est une tâche complexe, puisque nous n'avons pas de canal de transmission sécurisé à disposition. La solution est de mettre cette liste à jour à chaque maintenance du transpondeur, ce qui est malheureusement relativement peu fréquent. Entre temps, les dernières clés utilisées malicieusement ne pourront pas être ignorées. Une autre solution consisterait à charger manuellement la dernière CRL sur une carte SD ou un autre type de mémoire externe, puis de l'insérer dans le transpondeur comme on le ferait pour mettre à jour la dernière version des cartes pour le GPS. Cela ne serait cependant possible que pour des contrôleurs capables de lire de telles mémoires externes.

Il est important de noter que le problème de mise à jour de la CRL ne concerne que les avions équipés d'ADS-B-*in*. Les contrôleurs ont eux accès en permanence à la version actualisée de la liste fournie par l'OACI. Cela limite grandement l'ampleur d'une attaque avec une clé volée récemment, car celle-ci n'affectera pas l'ATC. Un message radio pourra de plus, en zone contrôlée, être diffusé aux avions pour les informer de l'attaque en cours.

4.2.2.3 Certificat à court terme

L'utilisation d'un certificat à court terme est complexe, car elle nécessite la réception du certificat avant chaque vol. Elle ne sera nécessaire que lors de certaines situations de vol définies par la NAA du pays concerné, dans des cas où une sécurité accrue est nécessaire. En effet ce type de certificat apporte un certain nombre d'avantages. Tout d'abord, le problème des CRL n'est plus présent, puisque le certificat n'est valide que pour 48H. Cela laisse très peu de temps à un attaquant pour voler les clés nécessaires puis mettre en place son attaque.

De plus, le certificat est produit pour un vol donné. Cela signifie une période précise, mais possiblement également une zone précise. En effet, la NAA pourrait mettre en place des CA locales, dont les certificats ne seraient valables que pour une zone de vol réduite. Un avion possédant un certificat à court terme a donc forcément été autorisé à voler à cet instant, et dans la zone du CA qui lui a délivré le certificat. Dans le cas du certificat à long terme, l'avion pouvait à tout moment se certifier une nouvelle chaîne de clés, sans avoir été autorisé à voler. De plus, limiter la validité du certificat à une zone aurait signifié empêcher l'avion

de voler autre part, non pas uniquement pour un vol, mais pendant toute l'année de validité du certificat. Ce n'est donc pas envisageable.

Certains pays n'ayant pas une aviation civile très développée n'auront pas la nécessité d'un tel niveau de sécurité, et utiliseront uniquement les certificats à court terme. Même aux États-Unis, l'utilisation généralisée de ce type de certificat engendrerait une surcharge de travail inutile pour la FAA. En effet, pourquoi vouloir par exemple limiter les autorisations de vol d'un avion qui volerait à vue (VFR) et n'aurait même pas besoin de remplir de plan de vol ? Cela provoquerait plus de complications pour la FAA et diminuerait plus les libertés de vol que cela n'améliorerait la sécurité aérienne. En revanche, nous pouvons citer certains cas pour lesquels un tel niveau de sécurité serait appréciable.

L'espace aérien d'un pays est généralement divisé en plusieurs zones dans lesquelles des règles différentes s'appliquent. Aux États-Unis, et de manière similaire au Canada, voler dans les espaces de catégorie A, B ou C requière la soumission d'un plan de vol et son acceptation par l'ATC. Ces zones se trouvent par exemple autour des grands aéroports, où le trafic est dense, ou à des altitudes élevées, là où se trouvent les couloirs aériens des avions de ligne. Il est important qu'un avion obtienne une autorisation avant de rentrer dans une telle zone, car c'est là que le risque de collision et ses conséquences sont les plus importants. De plus, un avion devrait pouvoir détecter immédiatement une attaque sans devoir attendre une alerte des contrôleurs, afin de pouvoir réagir plus rapidement.

Un autre cas d'utilisation des certificats à court terme serait lors du déclenchement de plans d'urgence tels que le Security Control of Air Traffic and Air Navigation Aids (SCATANA) aux États-Unis ou encore le Emergency Security Control of Air Traffic (ESCAT) au Canada. Ceux-ci requièrent l'atterrissage immédiat de tous les avions, excepté ceux autorisés tels que les secours ou la police. SCATANA a par exemple été utilisé après les attaques terroristes du 11 septembre 2001. Il est alors important de pouvoir instantanément distinguer les avions qui ont obtenu un certificat de vol spécial.

Dans le cas d'une situation où le certificat à court terme serait obligatoire, il faut un moyen de récupérer celui-ci. Une première possibilité est qu'un avion transmette à l'ATC sa clé publique en même temps que son plan de vol de façon sécurisée par internet avant le décollage. En retour, le certificat associé lui sera remis avec l'acceptation du plan de vol. Celui-ci sera transféré sur une mémoire externe, par exemple une carte SD, qui sera ensuite insérée dans le transpondeur, s'il possède un lecteur.

Si le transpondeur n'est pas capable de lire de mémoire externe, une autre solution doit être trouvée. De même, il arrive que des vols VFR, n'ayant pas transmis de plan de vol et ne possédant donc pas de certificat à court terme, veuillent rentrer dans des zones contrôlées dans

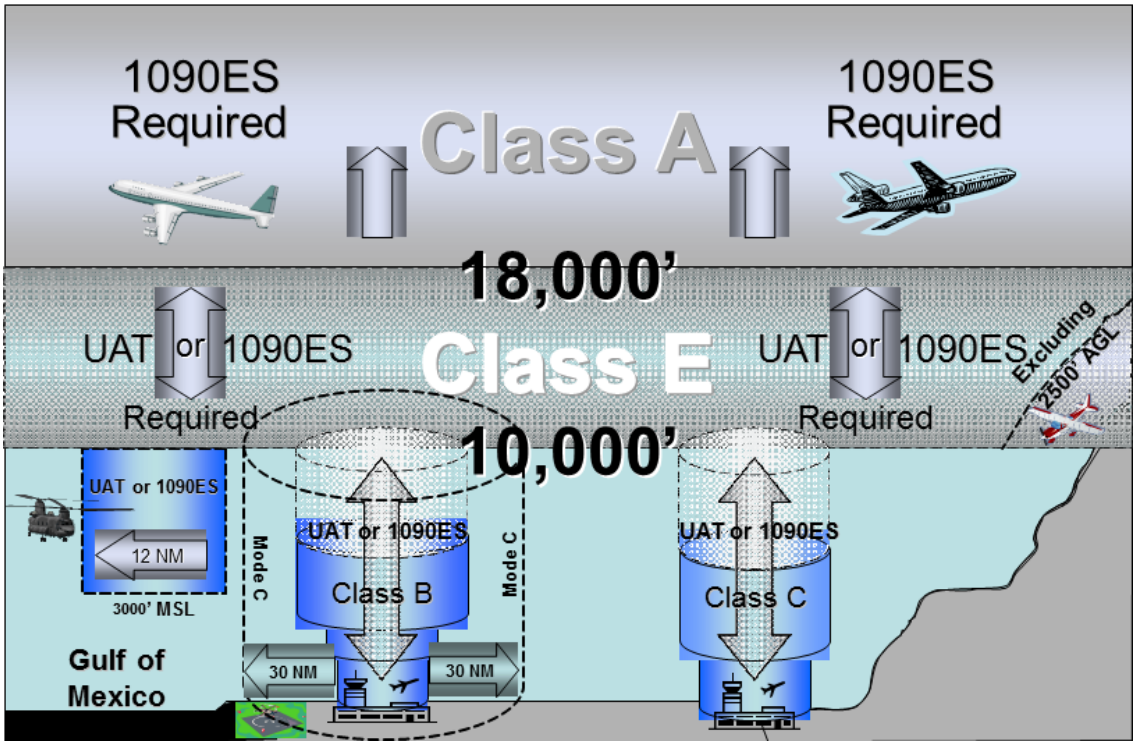


Figure 4.2 Les différentes classes d'espace aérien aux États-Unis. *Source : faa.gov*

lesquelles ceux-ci sont imposés. Dans ce cas-ci, une autorisation spéciale doit être demandée par radio aux contrôleurs. Nous proposons donc une procédure d'obtention du certificat à court terme, applicable dans les deux cas précédents, et représentée en figure 4.3 :

1. Dans un premier temps, que ce soit avant le décollage ou en cours de vol, le pilote contacte le contrôleur et l'informe qu'il désire obtenir un certificat à court terme.
2. Le contrôleur doit alors authentifier le pilote, Alice, et son avion, pour décider si oui ou non il accorde l'autorisation de vol et octroie en conséquence un certificat. Pour cela, il lui pose certaines questions auxquelles un attaquant ne serait pas supposé pouvoir répondre en totalité. Voici un certain nombre de questions possibles (les informations sont extraites du certificat d'immatriculation [Annexe F] et du dernier plan de vol [Annexe G]) :
   - Quel est le numéro d'immatriculation de l'avion ?
   - Quels sont le nom complet et l'adresse du propriétaire de l'avion tel qu'inscrit sur le certificat d'immatriculation ?
   - Quelle est la date d'immatriculation de l'avion ?
   - Des informations sur l'avion telles qu'inscrites au dernier plan de vol déposé. Quelle est sa marque ? Sa couleur ? Ses équipements de survie ?

   - Quelle est l'heure de dépôt du dernier plan de vol déposé ?
   - Donnez des informations sur la route du dernier plan de vol déposé.
   - Quelle était votre autonomie de vol pour le dernier plan de vol déposé ?
3. Si Alice a bien répondu à toutes les questions, et que son accès à la zone de vol qu'elle réclame ne cause pas de danger, le contrôleur commence la procédure d'émission du certificat. Il demande alors à Alice de choisir un nombre aléatoire.
4. Un paquet ADS-B spécifique sera réservé pour l'échange des certificats, tel que décrit en 4.2.3. Alice crée alors un message composé de sa clé publique et du nombre aléatoire qu'elle a choisi (ce dernier peut être rentré dans le transpondeur à l'aide de la molette de choix du code transpondeur). Elle chiffre ensuite ce message de façon asymétrique à l'aide de la clé publique de la NAA, et l'envoie au sein du paquet ADS-B.
5. Le contrôleur reçoit le message, et le déchiffre à l'aide de la clé privée de la NAA. Il prend connaissance de la clé publique et du code aléatoire.
6. Le contrôleur continue sa communication radio avec Alice, en lui demandant de lui communiquer le code aléatoire qu'elle avait choisi. Si le code qu'Alice indique est celui contenu dans le message reçu par le contrôleur, c'est que celui-ci provenait bien d'Alice. En effet, un attaquant qui aurait écouté la transmission radio et aurait envoyé un message ADS-B d'obtention du certificat au moment adéquat en essayant de se faire passer pour Alice n'aurait pas connu ce code.
7. Puisque la clé publique reçue provient bien d'Alice, le pilote émet au sein d'un paquet ADS-B le certificat à court terme correspondant à cette clé.

Notons que la clé privée du certificat à long terme de l'avion aurait pu être utilisée pour produire une signature de la clé publique qu'il veut utiliser pour son certificat à court terme. Cependant, mélanger de telle sorte l'utilisation des deux types de certificats n'est pas recommandable. En effet, un attaquant qui volerait la clé à long terme d'un avion pourrait alors obtenir des certificats à court terme pour ce même avion.

4.2.3 Envoi des certificats

Chaque avion est en possession de la totalité des clés publiques des sous-autorités de certification autorisées par l'OACI. Cela est possible, car leur nombre est réduit, et elles ne changent que peu fréquemment (dans le cas éventuel où une clé serait volée ou une NAA révoquée). Stocker directement les certificats de tous les avions n'est par contre pas réalisable. En effet, ceux-ci sont beaucoup trop nombreux, et changent régulièrement. Même pour les certificats à long terme, il n'est pas possible de fixer une date à laquelle tous les avions changeraient de

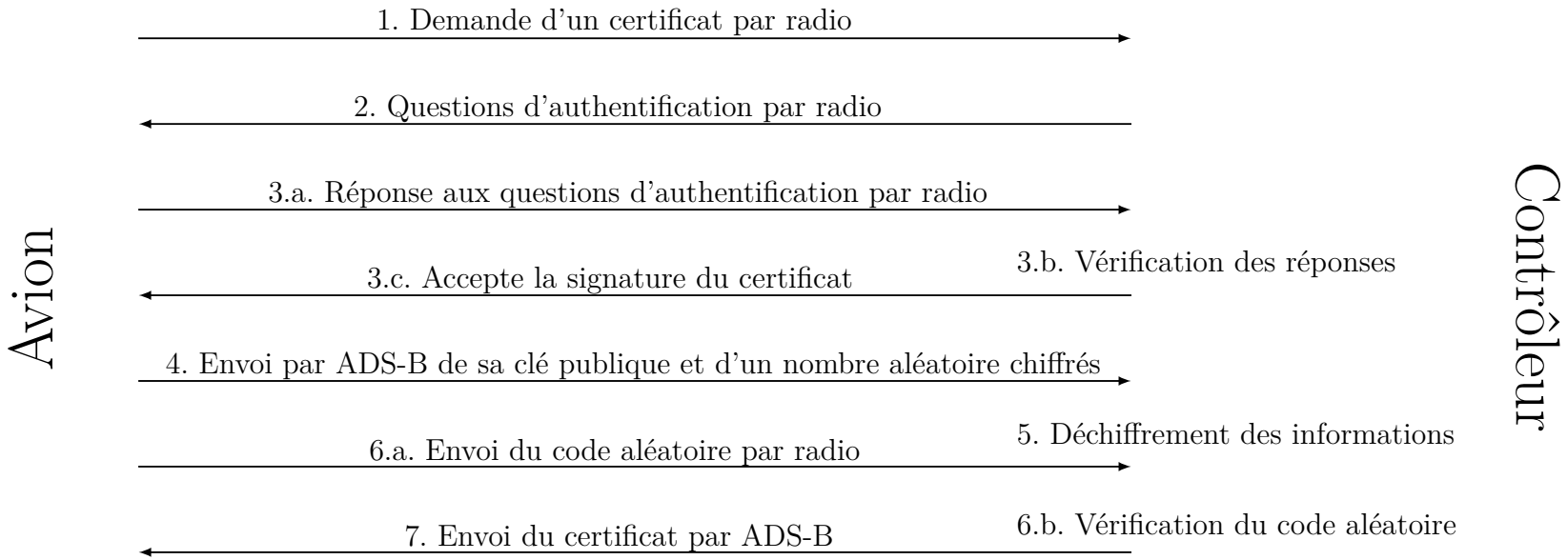


Figure 4.3 Procédure d'obtention du certificat à court terme

clé simultanément. Cela signifie qu'une partie des certificats est renouvelée tous les jours, et il n'est pas possible de mettre à jour les transpondeurs de façon aussi régulière pour y inclure la dernière liste de certificats, qui de toute façon ne rentrerait probablement pas en mémoire.

Puisque les certificats de tous les avions ne peuvent pas être stockés dans les transpondeurs, ils doivent donc être reçus en cours de vol pour permettre l'identification des messages d'un nouvel avion.

4.2.3.1 Envoi sur demande

Une première possibilité consiste à récupérer les certificats par l'intermédiaire de requêtes. Bob est en vol, et authentifie déjà les messages de plusieurs avions à proximité. À un moment donné, il croise la route d'un nouvel avion, Alice, dont il ne connaît pas le certificat. Il va alors diffuser un message demandant à Alice d'envoyer son certificat. Carole, qui vient également de recevoir un premier message d'Alice, entend la requête de Bob. Elle n'en émet pas de nouvelles, mais attend simplement qu'Alice diffuse son certificat pour le recevoir tout comme Bob.

Cette solution a comme avantage que le certificat d'Alice peut être reçu instantanément dès lors qu'un nouvel avion a besoin de le connaître. On s'éloigne en partie du fonctionnement *broadcast* de TESLA, puise l'on est en présence de requêtes. Cependant, la réponse d'Alice à une requête est destinée à tous les avions à proximité, et pas uniquement à l'émetteur de la requête. Cela n'est donc pas très problématique.

L'émission de requêtes a tout de même pour inconvénient de nécessiter l'envoi de messages supplémentaires, qui utilisent le peu de bande passante disponible. Cela reste négligeable en vol de croisière, lorsque l'on croise relativement peu d'avions. Mais imaginons le cas où Bob arriverait près d'un gros aéroport. Il ne connaît aucun des certificats des avions présents sur l'aéroport et à ses alentours, ce qui peut en représenter plusieurs dizaines. Il doit alors envoyer autant de requêtes, auxquelles doivent répondre chacun des avions, simultanément. Bob, sans le vouloir et pour une raison légitime, vient de saturer la bande passante. Il y a de fortes chances que, même si elles sont émises avec des délais aléatoires, les réponses des avions rentrent en collision et soient perdues. Un attaquant pourrait alors bien entendu causer un déni de service en émettant de nombreuses requêtes de certificats auprès de tous les avions présents.

4.2.3.2 Diffusion périodique

Pour cette raison, nous choisissons plutôt pour la suite une émission périodique des certificats. Chaque avion émettra, à un intervalle prédéterminé, son certificat. Le choix de la durée de cet intervalle est important, et sera discuté en détail en 5.4.

Lorsqu'il arrive à portée d'Alice, Bob doit attendre que cette dernière diffuse son certificat pour être capable de commencer à authentifier ses messages. Un intervalle trop long retarde donc d'autant le processus d'authentification. À l'inverse, un intervalle trop court demandera l'émission de beaucoup de messages et utilisera une trop grande quantité de bande passante.

4.2.3.3 Utilisation des signatures basées sur l'identité

Une dernière possibilité est l'utilisation de signatures basées sur l'identité (IBS). Le concept a déjà été abordé en 2.4.2.6. L'objectif était alors que la clé publique d'un avion puisse être extraite de différentes informations le concernant (numéro d'immatriculation, identifiant 24 bits de l'OACI, ...). Cependant, la clé privée associée était directement utilisée pour produire des signatures pour chaque message envoyé, ce qui utilisait trop de bande passante.

L'idée serait ici de plutôt se servir de cette clé privée pour produire les signatures de la chaîne de clés de l'avion. Ainsi, lorsque Bob croise la route d'Alice, il n'a qu'à écouter les messages qu'elle envoie. Il peut alors connaître les différentes informations sur son identité dont il a besoin pour produire sa clé publique. À partir de celle-ci, il sera capable de vérifier la signature produite par Alice pour sa chaîne de clés, et ainsi authentifier ses messages.

La clé privée est fournie à Alice par la NAA. La NAA est la seule à pouvoir générer cette clé, grâce à sa propre clé privée et aux informations sur l'identité d'Alice. Cependant, l'identité telle que présentée en 2.4.2.6 ne change pas au cours du temps. La méthode peut alors s'apparenter à un certificat à long terme, mais non pas à court terme. Pour cela, il faudrait ajouter certaines informations à l'identité d'Alice, caractéristiques du vol en cours. Celles-ci pourraient être extraites du plan de vol, dans le cas où celui-ci existe. On pourrait par exemple choisir des informations sur la route empruntée (*waypoints*, nom de code des prochaines étapes du vol), pour limiter le vol à la zone correspondante, ou encore les heures de départ et d'arrivée prévues pour le limiter dans le temps. Ces informations devraient alors être diffusées par ADS-B en cours de vol pour que la clé publique correspondante puisse être générée. La spécification de l'ADS-B contient déjà des types de messages pouvant contenir de telles informations, même s'ils ne sont généralement pas utilisés actuellement. L'annexe D présente par exemple un message contenant les prochains *waypoints* du vol.

Nous n'étudierons pas plus en détail cette solution au cours de ce mémoire, mais elle mé-

riterait d'être approfondie lors de recherches futures. En effet, son implémentation soulève plusieurs questions auxquelles il faudrait répondre. Quelles seraient les informations les plus appropriées pour la génération des clés ? Le surcoût de leur transmission par ADS-B serait-il inférieur à celui de la transmission directe de la clé et de son certificat ? Quel algorithme choisir pour la génération des clés, et quel en serait le niveau de sécurité ? Pour la suite, nous considérerons la diffusion périodique des certificats. Cependant, SAT pourrait facilement évoluer pour y intégrer une solution d'IBS.

4.2.4 Certification de la chaîne de clés Tesla par l'avion

L'avion d'Alice possède désormais un certificat à *court* ou *long* pour sa paire de clés publique K_{pub} et privée K_{pr}. Ce certificat est diffusé régulièrement de telle sorte qu'il puisse être vérifié. Dans la version traditionnelle de TESLA, K_{pr} était utilisée afin de produire une signature pour la clé K_0 (en bout de chaîne, mais la première à être utilisée et dévoilée). Bob, lorsqu'il veut authentifier pour la première fois un message M_j d'Alice produit à l'intervalle i, attend de recevoir le certificat d'Alice, puis K_0 et la signature correspondante. Il peut alors utiliser K_{pub} pour vérifier cette signature et K_0. Il doit ensuite calculer i hashs pour s'assurer que $K_0 = F^i(K_i)$. Lorsque l'on passe à l'intervalle suivant, le processus sera plus rapide puisque Bob n'aura à calculer qu'un seul hash pour s'assurer que $K_i = F(K_{i+1})$.

Nous avons choisi d'utiliser une chaîne de clés d'une durée de vie de 48H. Cela est relativement court, mais selon la durée T_{int} de l'intervalle peut tout de même représenter plusieurs milliers de clés. Comme expliqué en 4.2.2.1, cela signifie autant de milliers de hashs à calculer lors de la première authentification. Le calcul d'un hash est relativement court, mais leur cumul peut tout de même introduire un certain délai.

Pour corriger cela, nous allons utiliser la capacité qu'a un avion de certifier ses propres chaînes de clés. Cependant, nous ne souhaitons pas certifier régulièrement de nouvelles chaînes courtes, ce qui casserait la résistance aux pertes de paquets pour l'authentification de messages des chaînes antérieures. Nous choisissons plutôt de régulièrement certifier la dernière clé diffusée, K_{i-d}. Ainsi, à la réception du certificat, Bob pourra instantanément remonter la chaîne de clés d'Alice jusqu'à K_{i-d}. Il n'aura plus qu'à vérifier que $K_{i-d} = F^d(K_i)$. Le calcul de i hash initial a été réduit a seulement d hashs, ce qui est bien moindre.

Le certificat d'Alice n'est toutefois pas forcément diffusé à chaque intervalle, ce qui utiliserait trop de bande passante, comme vu précédemment. La dernière clé diffusée ne sera donc pas forcément connue. En revanche, il y aura toujours un gain de performance, puisque la dernière clé connue sera plus récente que K_0. La vérification de K_i nécessitera donc le calcul de moins de hashs que s'il avait fallu redescendre jusqu'à K_0.

4.3 Numéros de séquence contre les attaques par rejeu

Il reste un problème de sécurité avec l'utilisation de TESLA telle que nous l'avons présentée jusqu'à présent. La même clé reste utilisée pour tous les messages envoyés pour la durée d'un intervalle. Oscar, un attaquant, ne peut pas produire le MAC d'un nouveau message pendant cet intervalle, puisqu'il ne connaît pas la clé. En revanche, il pourrait choisir de retransmettre un message qu'il a déjà reçu au cours de l'intervalle. Puisque le message est le même, et que la clé est toujours valide, le MAC du message le sera également. Cela est problématique, car la vitesse d'un avion peut atteindre 500 nœuds, soit plus de 250 $m.s^{-1}$. L'information émise au début de l'intervalle peut donc se retrouver très erronée à la fin de celui-ci.

Cette possibilité d'attaque est empêchée en ajoutant un numéro de séquence pour chaque message. Celui-ci sera réinitialisé au début de chaque intervalle, puisque les messages précédents ne pourront alors plus être réutilisés (la clé aura changé). Si un message valide est réémis avec le même numéro de séquence qu'un ancien message du même intervalle, il sera ignoré par le récepteur. De plus, puisque le numéro de séquence sera pris en compte pour le calcul du MAC, un attaquant ne pourra pas se contenter de reprendre un ancien message et d'en modifier le numéro de séquence, puisque cela rendra son MAC invalide.

Un point important à noter est qu'il est nécessaire de conserver et de stocker dans la file d'attente tous les messages reçus, même s'ils ont le même numéro de séquence. L'élimination des messages ayant le même numéro de séquence se fera à la validation, une fois la clé reçue. En effet, autrement un attaquant pourrait provoquer une attaque par dénis de service en envoyant très peu de messages. Il suffirait qu'il prenne l'identification d'un avion donné, et envoie plus rapidement que ce dernier des messages avec de nouveaux numéros de séquence. Ceux de l'émetteur légitime paraîtraient alors en doublon et seraient éliminés. Une attaque par dénis de service est toujours possible en remplissant le tampon du récepteur de messages. Cependant, comme expliqué en 3.2, la bande passante serait probablement saturée avant le tampon du récepteur.

4.4 Niveau d'authentification d'un message

Nous avons vu que l'authentification des messages d'un avion était sujette à certains délais. C'est la contrepartie de l'utilisation de TESLA. C'est en effet ce délai qui nous permet d'obtenir un processus d'authentification asymétrique tout en utilisant des clés symétriques, qui limitent l'utilisation de la bande passante.

À un instant donné, les informations qui sont à notre disposition nous permettent d'assigner un certain niveau de confiance à un message reçu. Ces différents niveaux sont représentés par

des codes de couleur, par ordre croissant de confiance : rouge, orange, jaune et vert. Ceux-ci sont présentés à la figure 4.4 et ont les significations suivantes :

Rouge Après avoir reçu le certificat d'Alice et la clé d'un intervalle précédent de sa chaîne, Bob s'aperçoit que le MAC d'un message envoyé dans cet intervalle est incorrect. Ce message est alors classifié comme *rouge*. Cela signifie qu'Oscar, un attaquant, a très probablement essayé d'envoyer ce message avec un MAC aléatoire, puisqu'il ne connaissait pas la clé. L'autre possibilité est qu'il y ait eu une erreur lors de la transmission, mais cela est peu probable. En effet, si le message a été conservé, c'est que le code de vérification du message (CRC) était valide. L'erreur ne se serait donc produite que sur les bits du MAC. Si plusieurs messages successifs sont rouges, Bob pourra contacter les contrôleurs pour leur demander des consignes, ou bien suivre une procédure appropriée qui aura été définie pour ce cas de figure.

Orange Bob vient de recevoir un premier message provenant supposément d'Alice, dont il croise le chemin pour la première fois et ne connaît donc pas le certificat. Puisqu'il n'a encore jamais pu authentifier de message d'Alice, il ne sait pas si elle est réellement en vol à proximité. Oscar, l'attaquant, pourrait être en train d'usurper son identité pour produire un avion fantôme. Ce type de message est qualifié d'*orange*. Dès lors qu'un premier message aura été authentifié, les messages subséquents dont on ne connaît pas la clé deviendront *jaunes*.

Jaune Bob connaît le certificat d'Alice et a déjà pu authentifier au moins un de ses messages précédents, produit à un intervalle antérieur, mais relativement récent. Il sait donc qu'Alice est réellement en train de voler à proximité. Cependant, il n'a pas encore reçu la clé nécessaire pour vérifier le MAC d'un message produit à un intervalle plus récent. Ce message est qualifié de *jaune*. Un message jaune pourrait provenir d'Oscar, qui chercherait à modifier la trajectoire d'Alice telle que perçue par Bob. Un saut aberrant entre la dernière position authentifiée et celle indiquée par le message jaune peut toutefois être rapidement repéré. De plus, un message en provenance d'Oscar deviendra rouge dès lors que la clé de l'intervalle sera dévoilée. Bob saura donc qu'il devra se méfier des messages jaunes subséquents, qui auront alors une forte probabilité de provenir également d'Oscar.

Vert Un message *vert* a été authentifié par Bob. Le certificat d'Alice est connu, et la clé qui a servi à produire le MAC du message a été dévoilée. On est donc certain de son contenu (dans la limite de la sécurité de la fonction de hachage et de la taille du hash que nous avons choisis). Cependant, un message vert a forcément été produit à un intervalle antérieur, et est donc un peu daté. Il est toutefois très important, car il sert à authentifier l'historique de la

trajectoire d'Alice ; il permet de juger si les messages jaunes subséquents reçus sont en accord avec cette trajectoire et sont donc probablement authentiques.

4.5 Authentification anticipée des messages par les contrôleurs

Nous venons de voir que l'utilisation de SAT ne permet d'authentifier les messages ADS-B qu'avec un certain délai, dont la durée sera étudiée au chapitre 5. Entre temps, il n'est possible que de leur accorder un niveau de confiance *orange* ou *jaune*.

Cela est vrai pour l'utilisation de l'ADS-B-*in* par un autre avion. En revanche les contrôleurs, eux, n'ont pas à attendre la diffusion du certificat d'Alice. Ils ont directement accès à une base de données de tous les certificats produits par la NAA de leur pays et même, par l'intermédiaire d'un registre global géré par l'OACI, à ceux délivrés par les autres pays. Cela élimine le délai introduit par la réception des certificats et donc les messages orange.

En revanche, puisque chaque avion est chargé de produire et signer ses propres chaînes de clés SAT, les contrôleurs ne connaissent *a priori* pas à l'avance les clés utilisées pour produire les MAC. Le délai introduit pour l'authentification des messages par TESLA reste donc présent, ainsi que les messages *jaunes*. Alors que cela est convenable pour la réception en ADS-B-*in*, qui est seulement une aide à la navigation, un délai trop long pour l'authentification pourrait ne pas être satisfaisant pour les contrôleurs, qui sont les vrais responsables de la sécurité des vols. Comme expliqué précédemment, un pilote qui a des doutes sur les messages jaunes qu'il reçoit devrait pouvoir se tourner vers les contrôleurs. Ceux-si doivent donc à tout moment être certains de l'authenticité des messages ADS-B reçus, avec un délai réduit.

Dans le cas où le délai que nous obtiendrons en 5.6, qui sera le délai le plus court possible respectant toutes les autres contraintes de l'ADS-B, ne serait pas suffisant du point de vue des contrôleurs, une solution est possible. Celle-ci consiste à transmettre avant le décollage toute la chaîne de clés SAT du vol aux contrôleurs. Ainsi, puisque toutes les clés sont alors connues en avance, il devient possible pour les contrôleurs d'authentifier instantanément les messages. En ce qui concerne la transmission de la chaîne de clés, celle-ci pourrait se faire de façon chiffrée grâce à la clé publique de la NAA, comme proposé en 4.2.2.3 pour la transmission de la clé publique de l'avion et du code aléatoire.

Nous ne proposons pas cette solution par défaut, car elle suppose le stockage et la diffusion, à tous les contrôleurs de tous les pays, de la chaîne de clés de vol de chaque avion. La possibilité de fuite des clés est donc forte, et il est préférable d'avoir un petit délai à l'authentification plutôt qu'une divulgation massive des clés, qui rendrait une attaque très facile et inévitable. Une telle proposition pourrait tout de même être utilisée dans des cas

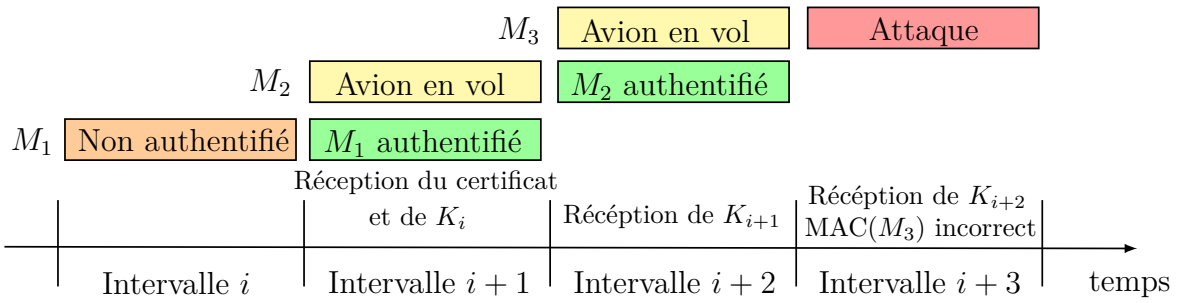


Figure 4.4 Les différents niveaux d'authentification d'un message. M_1 est reçu à l'intervalle i, alors que le certificat de l'émetteur Alice n'est pas encore connu. M_1 est donc *orange*. À l'intervalle $i+1$, le certificat d'Alice et K_i son reçus, M_1 est donc authentifié et devient *vert*. M_2 est alors reçu. Il ne peut pas être authentifié, mais on est certain qu'Alice est en vol à proximité, puisque M_1 a été authentifié. M_2 est donc *jaune*. À l'intervalle $i+2$, on reçoit K_{i+1}, M_2 est authentifié et devient *vert*. On reçoit également M_3 qui est alors *jaune*. À l'intervalle $i+3$, on reçoit K_{i+2}. On peut alors vérifier le code d'authentification de M_3, mais celui-ci n'est pas valide. M_3 devient donc *rouge*, et la probabilité d'une attaque est élevée.

spécifiques. Par exemple, lors du déclenchement du plan SCATANA, le risque d'attaque est très fort. Il est donc important pour les contrôleurs de connaître la validité des messages en temps réel. Dans ce cas précis, les avions de secours ne volent que localement. Il serait donc possible de ne diffuser les clés d'un avion qu'aux contrôleurs d'une zone restreinte, ce qui limiterait grandement les risques de fuite de ces clés.

4.6 Messages utilisés pour la transmission des informations d'authentification

L'authentification des messages ADS-B d'Alice grâce à SAT requiert la transmission d'informations supplémentaires, non définies dans le protocole initial :

- Le MAC de chaque message.
- La clé publique d'Alice transmise à la NAA en échange du certificat correspondant.
- La clé publique d'Alice et le certificat correspondant transmis aux autres avions.
- Une clé de la chaîne servant à authentifier cette dernière et la signature correspondante.
- La clé d'un intervalle antérieur dévoilée régulièrement.
- Le cas échéant la transmission de toute la chaîne de clé de façon anticipée aux contrôleurs.

En ce qui concerne le MAC d'un message, nous choisissons de simplement l'envoyer à la suite de ce message, grâce à des bits supplémentaires. Un ancien transpondeur, configuré pour recevoir des messages ADS-B de 112 bits, ignorera simplement les bits supplémentaires.

En ce qui concerne les autres informations, puisqu'il n'existe aucun message ADS-B existant capable de les contenir, nous choisissons de créer un nouveau type de message. Les messages ADS-B traditionnels ont le format 17, et de nombreux autres formats sont déjà utilisés (19 pour un usage militaire, 0 ou 16 pour le Traffic Collision Avoidance System (TCAS), ...). Le format 14 est en revanche actuellement inutilisé. C'est donc celui que nous choisissons, et il sera ignoré des anciens transpondeurs ne sachant pas qu'il correspond à SAT.

CHAPITRE 5 CHOIX DES PARAMÈTRES DE SAT

Nous avons présenté au chapitre 4 les différentes adaptations que SAT apporte à TESLA afin d'être en mesure de sécuriser l'ADS-B. Il nous reste cependant plusieurs choix techniques à faire quant aux choix des différents paramètres de SAT. Ceux-ci doivent être choisis de façon optimale de manière à rendre le protocole le plus performant possible, notamment en matière de délai d'authentification, qui doit être le plus court possible. Cela doit cependant être fait tout en prenant en compte les contraintes de l'ADS-B, notamment en termes de bande passante disponible.

5.1 Taille des clés

Le premier élément qui caractérise la sécurité d'un protocole cryptographique donné est la taille des clés utilisées. En ce qui concerne SAT, deux types de clés sont utilisées : des clés asymétriques pour la production des certificats d'avions et des clés symétriques pour les chaînes TESLA. Nous allons donc étudier la taille des clés qui sera la plus appropriée dans chacun des deux cas.

5.1.1 Clés asymétriques pour les certificats des avions

La certification des avions et de leurs clés est faite de façon asymétrique par les NAA de chaque pays. Ces dernières ont auparavant elles-mêmes obtenu un certificat de la part de l'OACI, les désignant comme sous-autorités de certification. Nous proposons de produire la signature de ces certificats grâce à ECDSA. En effet, comme nous l'avons vu en 2.4.2.4, l'utilisation de la cryptographie sur courbe elliptique permet l'utilisation de clés plus courtes. Les signatures, quant à elles, restent de même taille que pour DSA, basé sur le problème du logarithme discret. Elles sont en revanche plus courtes que celles produites avec RSA, basé sur le problème de la factorisation de grands nombres premiers.

Comme les évolutions sont très longues dans le domaine aéronautique, nous souhaitons que la longueur de clés choisie par SAT reste suffisante au-delà de 2030. Le tableau 5.1 résume les recommandations de différents chercheurs et organismes gouvernementaux quant à la taille de clé nécessaire après 2030. Notons que les certificats à long terme auront seulement une durée de validité d'un an, et les clés des NAA seront également changées périodiquement pour plus de sécurité. Nous choisissons donc de suivre les recommandations de l'Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI, 2014). Les préconisations de cet organisme

gouvernemental français sont en effet dans la moyenne des valeurs recommandées. Nous ne choisissons pas la taille de clé la plus élevée préconisée par ECRYPT II (2012), car dans le cas très peu probable où une clé utilisée maintenant serait découverte après 2030, cette clé aurait déjà été remplacée et ne serait plus valide. L'économie de bande passante étant un enjeu important, il ne nous semble pas judicieux de surévaluer les risques.

Nous choisissons donc d'utiliser des certificats basés sur des signatures ECDSA produites avec des clés de 256 bits. Cela correspond à un niveau de sécurité de 128 bits, et les signatures produites ont une taille de 512 bits.

5.1.2 Clés symétriques pour les chaînes TESLA

Une fois son certificat obtenu, l'avion d'Alice est capable de signer sa propre chaîne TESLA. Nous devons désormais déterminer la taille des clés de cette chaîne. Becker et al. (2009) et Lo et Enge (2010) discutent largement ce choix pour l'application de TESLA à la sécurisation du GNSS, et leur discussion reste largement valide dans le cas de SAT.

Deux types d'attaques sont en réalité possibles. Supposons que le délai d avant la diffusion d'une clé est d'un intervalle, que les clés sont d'une longueur de s bits et que la fonction de hachage utilisée pour produire la chaîne est $F : \{0,1\}^s \rightarrow \{0,1\}^s$. La clé K_{i-1} sera donc dévoilée au début de l'intervalle i. La première attaque possible est une attaque par *préimage*. Celle-ci consiste à trouver une clé valide pour l'intervalle i, c'est-à-dire K_v telle que $K_{i-1} = F(K_v)$. On peut avoir $K_v = K_i$, mais cela n'est pas nécessaire pour que K_v soit considérée comme une clé valide de la chaîne. En considérant que F est une fonction à sens unique parfaite, les images qu'elle produit sont équiprobables. La probabilité qu'une clé choisie aléatoirement soit valide est de $\frac{1}{2^s}$. Il faut donc en moyenne choisir $\frac{2^s}{2} = 2^{s-1}$ clés avant d'en trouver une valide. Puisque la clé de l'intervalle i n'est valable que pour la durée de cet intervalle, un attaquant doit alors être capable de calculer plus de 2^{s-1} hashs par intervalle pour avoir la chance d'obtenir une clé valide.

Un deuxième type d'attaque consiste à s'attaquer à la chaîne de clés plutôt qu'à une clé en particulier. En effet, en choisissant une durée d'intervalle relativement courte, il est très peu probable qu'un attaquant soit capable de calculer un nombre suffisant de hashs dans les temps. En revanche, plutôt que de repartir à chaque fois d'une nouvelle clé et d'y appliquer F pour vérifier si le résultat produit est le hash de la dernière clé publiée, une idée plus astucieuse consiste à choisir une unique clé K_c de manière aléatoire. On va alors appliquer F de façon répétée à $K_c : K_j = F^j(K_c)$. Deux possibilités justifieront l'arrêt du calcul. Dans un premier cas de figure, le calcul de F nous donne une clé déjà produite. Nous sommes alors tombés dans une boucle et il faut choisir un nouveau K_c puis recommencer. Dans un

Tableau 5.1 Taille des clés nécessaire pour que la cryptographie reste sûre au-delà de 2030
Source : BlueKrypt Giry et Quisquater (2015)

Méthode	Sûr juqu'en	Symétrique	RSA	Courbe elliptique
Lenstra/Verheul	2050	109	4047	206
Lenstra actualisée	2050	102	2440	203
ECRYPT II	> 2041	256	15424	512
NIST	$\gg 2030$	192	7680	384
ANSSI	> 2030	128	3072	256

second cas, on tombe sur une clé déjà publiée. On a alors gagné, puisque l'on a produit une chaîne de clés non encore dévoilées qui se connecte à la véritable chaîne de clés d'Alice. Il sera alors possible de falsifier des messages pendant toute la durée de la chaîne que nous avons produite.

La chaîne produite dans chacun des deux cas précédents a une forte probabilité d'être longue. En effet, tomber soit sur une boucle, soit sur une clé déjà publiée constitue une collision sur notre fonction F. Le *paradoxe des anniversaires* indique qu'il faut en moyenne $2^{\frac{s}{2}}$ essais pour produire une collision, ce sera donc également la longueur moyenne de la chaîne produite. En revanche, la probabilité que la chaîne produite se rattache à la chaîne d'Alice est - comme pour le premier type d'attaque - égale à 2^{s-1} (les autres chaînes sont des boucles). En effet, pour chaque nouveau calcul de F - y compris celui ayant causé la collision -, la probabilité d'obtenir la dernière clé publiée est toujours de $\frac{1}{2^s}$.

Pour réduire le risque d'attaque réussie, il est possible de modifier légèrement le calcul de la chaîne de clés. Nous proposons de remplacer le calcul $K_i = F(K_{i+1})$ par $K_i = F(K_{i+1}, i)$. Ainsi, un attaquant ne peut plus juste choisir une clé aléatoirement et y appliquer F en espérant finir par retomber sur une clé d'Alice. Il doit en plus choisir une position pour sa clé initiale. Même s'il finit par retomber sur une clé d'Alice, il faut que ce soit au bon intervalle. Autrement, la clé ne sera pas vérifiée par Alice.

S'il choisit de commencer à l'intervalle $i + l$ - Alice se trouvant alors dans l'intervalle i -, l'attaquant a besoin de l calculs de hashs pour vérifier si son choix de K_{i+l} était bon, ce qui a toujours une probabilité de $\frac{1}{2^s}$ de se produire. Il faut donc en moyenne à l'attaquant $l * 2^{s-1}$ opérations pour trouver une chaîne valide de taille l. Pour que des clés de la chaîne soient encore valides une fois cette dernière trouvée, il faut que les $l * 2^{s-1}$ hashs aient été calculés en moins de l intervalles. On retombe alors sur le même résultat que pour le premier

cas d'attaque, à savoir que l'attaquant doit être capable de calculer plus de 2^{s-1} hashs par intervalle pour avoir la chance d'obtenir une clé valide.

En conclusion, avec le calcul de la chaîne de clés modifiée pour y intégrer le numéro de l'intervalle, il suffit de choisir une taille de clés ne permettant pas sa découverte pendant la durée d'un intervalle. Cependant, le choix d'une taille trop courte permettrait à l'attaquant de précalculer et stocker en mémoire un nombre important de chaînes possibles. La découverte d'une clé serait alors instantanée si celle-ci se trouve au sein d'une chaîne en mémoire. De plus, la découverte future de problèmes de sécurité dans la fonction à sens unique que nous avons choisie pourrait diminuer le nombre d'opérations nécessaires pour découvrir une clé. Pour ne pas prendre de risque, nous choisissons la taille de clés symétriques recommandée par l'ANSSI au-delà de 2030, soit 128 bits. Cela est un compromis nécessaire sur l'utilisation de bande passante pour s'assurer que SAT restera sécurisé sur le long terme.

5.2 Taille du code d'authentification

Un paramètre important est celui de la taille du message d'authentification (MAC). En effet, celui-ci se retrouve à chaque message. Il est donc important d'en optimiser la taille.

Nous choisissons tout d'abord d'utiliser une fonction de hachage de la famille SHA-2, celle-ci n'ayant à l'heure actuelle été compromise par aucune attaque connue. Comme expliqué en 2.4.2.2, elle sera utilisée pour la production d'un HMAC. C'est une méthode de génération d'un MAC - à partir d'une fonction de hachage et d'une clé - n'étant pas vulnérable à l'attaque de l'extension :

$$HMAC_K(m) = \text{SHA-2}\left((K' \oplus opad) \parallel \text{SHA-2}\left((K' \oplus ipad) \| m\right)\right) \tag{5.1}$$

Parmi les fonctions SHA-2, il en existe deux types : SHA-256, qui fonctionne sur des blocs de 512 bits et produit un hash de 256 bits, et le SHA-512 qui fonctionne sur des blocs de 1024 bits et produit un hash de 512 bits. Des variantes existent, semblables aux précédentes, mais fournissant des hashs tronqués. Nous choisissons de nous concentrer sur SHA-256. En effet, un hash de 256 bits est déjà très grand en comparaison des 112 bits d'un message ADS-B.

En réalité, il est très difficile pour un attaquant de falsifier un MAC. Puisqu'il ne connaît pas la clé, sa seule possibilité est de générer des MAC aléatoirement. Pour un hash de r bits, ses chances sont donc de $\frac{1}{2^r}$. En règle générale, la taille du hash doit être choisie en fonction des capacités de calcul de l'attaquant et de la durée pendant laquelle nous souhaitons que

notre MAC reste sûr. En effet, il faut que l'attaquant vérifie un à un les MAC aléatoires qu'il produit pour finalement trouver le bon après en moyenne 2^{r-1} essais.

Cependant, dans notre cas, l'attaquant n'a aucun moyen de vérifier si le MAC qu'il a produit est valide. En effet, la clé ne sera dévoilée qu'à un intervalle subséquent. De plus, la durée de l'intervalle pendant lequel la clé reste la même est courte. Il faudrait donc que l'attaquant diffuse tous les paquets qu'il produit avec des MAC différents. Leur nombre est forcément réduit par la faible bande passante disponible. De plus, au moment de la vérification, il suffit de quelques messages rouges (dont le MAC était invalide) pour déclencher une alerte. L'attaquant sera donc forcément identifié, et même s'il parvient à produire un message vert, celui-ci passera pour du bruit en regard de la trajectoire produite par les messages légitimes de l'émetteur. On peut donc se contenter d'une taille de hash relativement faible pour le MAC, ce qui économisera largement l'utilisation de bande passante tout en restant suffisamment sécuritaire.

Nous choisissons une taille de hash de 16 bits, ce qui correspond à 65536 possibilités pour une augmentation de seulement 14% de la taille du message.

5.3 Taille du numéro de séquence

La taille du numéro de séquence, tout comme celle du MAC, se retrouvera à chaque message et doit donc être la plus courte possible pour économiser l'utilisation de bande passante. Elle doit cependant tout de même être suffisante pour que chaque message envoyé au sein d'un même intervalle puisse avoir un numéro unique.

La durée de l'intervalle sera discutée précisément en 5.6, mais ne durera pas plus que quelques secondes afin de limiter le délai d'authentification. Un avion envoie actuellement rarement plus de 5 messages par seconde, comme décrit en 2.2 (deux messages par seconde pour la position et la vitesse, et parfois un pour l'identification). D'autres types de messages sont prévus dans la spécification de l'ADS-B, mais ne sont que peu fréquemment utilisés et ne le seront probablement pas plus dans le futur pour limiter l'utilisation de bande passante.

Nous choisissons donc une taille de 8 bits pour le numéro d'identification, ce qui représente 256 messages possibles par intervalle. Ce nombre présente une marge relativement grande par rapport à l'utilisation actuelle, et dans le cas exceptionnel où le nombre serait dépassé pour un intervalle donné, les messages non prioritaires pourraient être déplacés vers l'intervalle suivant.

5.4 Fréquence d'émission des certificats

Comme nous l'avons vu en 4.4, la durée comprise entre le moment où nous recevons le premier message d'un nouvel avion et l'instant où nous réussissons à authentifier celui-ci est la plus critique. En effet, le message est alors de catégorie *orange*, c'est-à-dire qu'il pourrait s'agir d'un avion fantôme, non présent dans le ciel.

La durée pendant laquelle un message peut rester orange est fonction de la fréquence d'émission des certificats. Si Alice arrive à portée de Bob juste après qu'elle ait émis son certificat, Bob devra patienter jusqu'à ce qu'Alice les émette à nouveau, au bout d'une période déterminée. Bien sûr, si le paquet contenant le certificat est perdu, Bob devra à nouveau patienter pendant une seconde période.

Il nous faut donc déterminer la fréquence optimale d'émission des certificats. Comme toujours, une fréquence plus élevée diminue le délai d'authentification des messages, mais augmente l'utilisation de la bande passante. Nous devons donc diminuer au maximum cette fréquence, tout en conservant un délai d'authentification raisonnable.

Plutôt que de décider arbitrairement quelle serait cette fréquence optimale, nous avons choisi de constituer un panel d'experts. Nous souhaitions également y inclure des contrôleurs aériens, mais ces derniers n'ont pas répondu à nos sollicitations. Nous avons en revanche réussi à réunir deux pilotes. Nous leur avons posé quatre principales questions :

— Quels sont selon vous les risques principaux associés à un paquet *orange* ?

— Dans quelles conditions de vol ces risques sont-ils les plus importants ?

— Ces risques sont-ils les mêmes du point de vue du contrôleur et du pilote ?

— Quelle est en conséquence la fréquence minimale d'émission des certificats que vous recommanderiez ?

Leurs réponses peuvent être résumées en deux points importants. Tout d'abord, un appel radio peut permettre de rapidement vérifier si un avion suspect est réel où non, en particulier si celui-ci n'a pas rempli de plan de vol. Le délai d'authentification n'est alors pas très important en condition normale d'opération. En revanche, un dimanche soir, à l'arrivée d'un gros aéroport, le trafic peut être très important. S'il fait nuit, et que les conditions météo sont mauvaises, la tâche des contrôleurs peut alors vite devenir ardue. Si une attaque survient à cet instant, il sera très difficile pour les contrôleurs de faire manuellement les vérifications nécessaires. La confiance dans les messages ADS-B est alors cruciale, avec un faible délai pour l'authentification. Heureusement, dans le cas de SAT, les contrôleurs ont accès à la base de données globale des certificats, et n'ont pas besoin d'attendre leur émission par l'avion.

Du point de vue des pilotes, la problématique est différente. À l'approche d'un aéroport, ceux-ci seront guidés directement par les contrôleurs. C'est plutôt en vol de croisière que des problèmes peuvent survenir, lorsque les contacts radio avec l'ATC sont moins fréquents. Si un avion apparaît soudainement dans la trajectoire de l'avion, une manœuvre d'évitement sera nécessaire. Mais que faire si l'avion est encerclé de toutes parts par d'autres avions, certains potentiellement réels et d'autres fantômes ? Si la visibilité est mauvaise, le pilote n'a aucun moyen de les distinguer. Il ne veut pas avoir à jouer à la roulette russe pour décider la manœuvre à effectuer, en espérant que l'avion dont il va croiser la trajectoire ne soit pas réel.

C'est là que SAT intervient, en permettant au pilote de distinguer les avions *verts* des avions *rouges*. La fréquence d'émission des certificats la plus faible acceptable correspond donc au temps qu'a un pilote pour choisir sa manœuvre d'évitement. Cela dépend en réalité de deux paramètres : la vitesse de l'avion et la portée de l'ADS-B. Un avion se déplaçant au maximum à 250 $m.s^{-1}$, deux avions volant face à face se rapprocheront donc de 500 mètres par seconde.

De plus, un avion ne peut pas apparaître instantanément sur les écrans à une distance très proche. Puisque la portée de l'ADS-B est de plusieurs centaines de kilomètres, un avion apparaîtra pour la première fois sur les écrans à cette distance. Autrement, on peut directement en déduire qu'il s'agit d'une attaque et que l'avion n'est pas réel. La portée maximale de l'ADS-B est d'environ 300 km, mais nous choisissons de considérer une portée de 100 km afin de conserver une large marge de sécurité.

À la vitesse de rapprochement de 500 $m.s^{-1}$, le pilote a donc 200 secondes afin d'effectuer sa manœuvre et d'éviter la collision, soit un peu plus de trois minutes. Sur ces trois minutes, les pilotes ont évalué qu'une durée de 30 secondes avant d'être certain que l'avion était réel - et de potentiellement commencer la manœuvre d'évitement - était acceptable. C'est donc la valeur que nous choisissons pour la fréquence d'émission des certificats.

5.5 Choix du délai avant la diffusion des clés

Le second type de délai avant l'authentification d'un message *jaune* correspond au temps entre la réception de ce message et la réception de la clé de l'intervalle correspondant. Ce délai dépend non seulement de la durée de l'intervalle, mais également du nombre d'intervalles d séparant l'utilisation d'une clé et sa diffusion, comme décrit en 3.1.2. Nous voulons choisir d le plus petit possible afin de limiter le délai, tout en respectant :

$$d > \frac{T_{trans} + \Delta}{T_{int}} \tag{5.2}$$

Nous avons vu en 4.1 que grâce au temps GNSS, la synchronisation entre les horloges de tous les avions était très précise. Nous avons pu obtenir $\Delta = 200\ ns$. En ce qui concerne T_{trans}, les paquets se propagent grâce à des ondes électromagnétiques, donc à la vitesse de la lumière de $3.10^8\ m.s^{-1}$. À la portée maximale de 300 km, cela correspond à $T_{trans} = 1\ ms$.

On constate que $T_{trans} + \Delta \gg T_{int}$, dont la durée sera de quelques secondes. On peut donc choisir $d = 1$. Une clé utilisée à un intervalle i sera diffusée à l'intervalle $i + 1$.

Il est toutefois important de souligner un point en particulier. Si Alice publie un message à la toute fin d'un intervalle i, il se peut que Bob soit déjà dans l'intervalle $i + 1$ au moment où il reçoit le message, à cause du temps de transmission. Il sera donc refusé. Deux solutions sont possibles pour contourner ce problème. La première consiste en ce qu'Alice n'envoie pas de message pendant la dernière milliseconde d'un intervalle. La seconde requiert qu'Alice ne diffuse pas sa clé immédiatement au début de chaque intervalle, mais uniquement après une milliseconde. Bob pourrait en contrepartie accepter les messages de l'intervalle i reçus lors de la première milliseconde de l'intervalle $i + 1$.

5.6 Durée de l'intervalle

Puisque nous avons choisi $d = 1$, le délai d'authentification pour les messages *jaunes* sera égal à la durée d'un intervalle, qu'il nous faut désormais définir. La durée de l'intervalle déterminera la fréquence à laquelle il sera nécessaire de diffuser une nouvelle clé de la chaîne, et donc influencera l'utilisation de bande passante.

5.6.1 Respect de la spécification de l'ADS-B

Jusqu'à présent, nous avons choisi les différents paramètres de SAT de façon à répondre de la meilleure manière possible aux besoins d'authentification. Nous fixions d'abord les critères de service minimum que nous souhaitions obtenir au niveau de SAT, puis seulement ensuite nous cherchions à optimiser l'utilisation de bande passante en respectant ces critères. Cependant, l'ADS-B possède également ses propres critères de service minimum, tels que décrits au sein de la spécification DO-242 de la RTCA (2002) et présentés au tableau 5.2. Ceux-ci doivent absolument être respectés pour que SAT puisse obtenir les certifications nécessaires à son déploiement. Nous devons donc cette fois-ci procéder de façon contraire. Nous

Tableau 5.2 Performances minimales requises pour l'ADS-B selon la spécification DO-242 définie par la RTCA (2002), du point de vue des contrôleurs. *Régional* signifie que l'avion se trouve en vol de croisière et *approche* qu'il est en phase de décollage ou qu'il se prépare pour l'atterrissage. Pour respecter la spécification, il faut que la probabilité de réception d'au moins un message d'une catégorie soit supérieure à 98% pendant la période de mise à jour. Cela est valable pour une certaine densité de trafic dans un rayon d'opération donné. L'utilisation d'une antenne 6 secteurs apporte un facteur d'amélioration de 2.5, qui équivaut à une diminution du nombre de messages reçus équivalente.

	Capacité opérationnelle		
Type de vol	Régional	Approche	Aérodrome
Rayon du domaine d'opération (NM)	200	60	5
Densité du trafic (avions)	1250	750	100 mobiles 150 immobiles
Période de mise à jour (s)	12	5	1
Probabilité de mise à jour réussie (%)	98	98	98
Type d'antenne	Omnidirectionnelle	6 secteurs	6 secteurs
Facteur d'amélioration γ	1	2.5	2.5

choisirons la durée minimale pour un intervalle qui permette de respecter la spécification de l'ADS-B, et discuterons ensuite l'acceptabilité du délai correspondant pour l'authentification des messages.

La RTCA-DO-242 définit trois situations distinctes dans lesquelles peut se trouver l'avion. Pour chacune d'entre elles, des capacités opérationnelles distinctes sont définies pour les contrôleurs et pour les transmissions air-air, c'est-à-dire pour les avions équipés de l'ADS-B *in*. La capacité opérationnelle nécessaire pour les contrôleurs est supérieure à celle air-air dans chacun des cas. C'est donc celle que nous présentons, le respect des capacités air-air en découlant.

Les trois situations à étudier sont les suivantes :

Régional ou encore *En Route* signifie que l'avion se trouve en vol de croisière. Comme ce type de vol requiert peu de communications avec les contrôleurs, un seul centre de contrôle s'occupe d'une large région, et gère une grande quantité d'avions de manière simultanée. C'est pourquoi le domaine d'opération et la densité du trafic nécessaire sont importants, respectivement de 200 milles marins (370 km) et 1250 avions. En revanche, comme il ne s'agit pas d'une phase critique de vol et que la séparation entre avions est importante, la

période à laquelle des informations de position actualisées doivent être reçues n'est que de 12 seconds, ce qui est relativement faible.

Approche concerne la zone proche d'un aérodrome, c'est-à-dire dans un rayon de 60 milles nautiques (110 kilomètres). De nombreux avions - jusqu'à 750 - se croisent. Ils s'apprêtent à atterrir, viennent de décoller, ou sont suffisamment proches pour qu'ils puissent rentrer en conflit avec des avions se trouvant dans les situations précédentes. Cette phase critique du vol requiert de recevoir de nouvelles informations de position au minimum toutes les 5 secondes.

Aérodrome concerne tous les avions au sol. Jusqu'à 150 avions peuvent se trouver stationnés, c'est-à-dire *immobiles*. Ils doivent cependant tout de même émettre de l'ADS-B pour pouvoir être repérés par les contrôleurs. 100 avions peuvent également être mobiles sur les pistes de l'aérodrome. Ils peuvent également être en train d'atterrir ou bien de décoller, dans un rayon de 5 milles nautiques (10 km) de l'aérodrome. Comme les espacements sont alors réduits et les risques de collision élevés, la mise à jour de la position doit être reçue au minimum toutes les secondes.

Dans chacune des trois situations énoncées, la période de mise à jour requise doit être respectée avec une probabilité supérieure à 98 %.

5.6.2 Modélisation et calcul de la capacité opérationnelle de SAT

Les premières simulations quant à la capacité opérationnelle du mode S datent du milieu des années 1990, et ont été effectuées au laboratoire *Lincoln* du Massachusetts Institute of Technology (Orlando et Harman, 1994). Elles consistent à modéliser le risque de collision entre deux messages par une loi de Poisson.

En effet, une loi de Poisson sert à étudier des événements se produisant de manière indépendante et à une fréquence moyenne de λ fois par unité de temps. Elle va alors nous permettre de calculer la probabilité que l'événement se produise exactement k fois à l'intérieur d'un intervalle de temps fixé t :

$$p(k) = \frac{(\lambda * t)^k}{k!} \exp(-\lambda * t) \tag{5.3}$$

Dans notre cas, la loi de Poisson va nous permettre de calculer la probabilité qu'aucun autre message ne soit envoyé pendant la période de vulnérabilité de notre message ADS-B, présentée

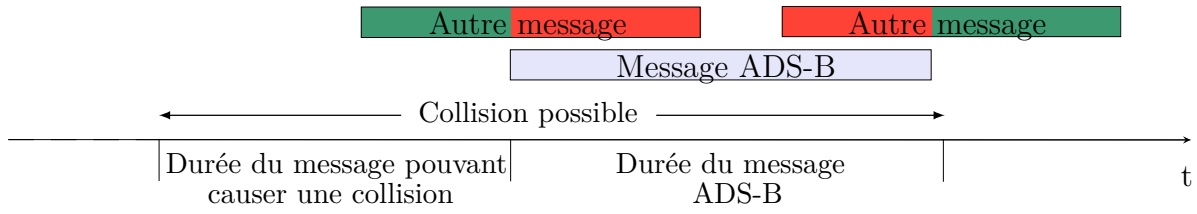


Figure 5.1 Temps pendant lequel une collision entre deux messages peut se produire

à la figure 5.1. Il y a collision si une partie d'un autre message est envoyé en même temps que le nôtre. La période de vulnérabilité est donc égale à la somme des durées des deux messages.

Si nous posons T_{ADS-B} la durée d'un message ADS-B, et respectivement T_j et λ_j la durée et la fréquence d'émission d'un message de type j, la probabilité qu'il n'y ait pas de collision entre les deux messages est :

$$p_j(0) = \exp\Big(- \lambda_j * (T_{ADS-B} + T_j)\Big) \tag{5.4}$$

Pour l'ensemble des types de messages émis sur la fréquence de 1090 MHz, la probabilité qu'il n'y ait aucune collision et que le message ADS-B soit reçu correctement est donc :

$$P_{réception} = \prod_j p_j(0) = \prod_j \exp\Big(- \lambda_j * (T_{ADS-B} + T_j)\Big) \tag{5.5}$$

L'utilisation d'antennes multi-secteurs dans les cas présentés au tableau 5.2 apporte un facteur d'amélioration γ quant au nombre de messages reçus par unité de temps. En effet, on remplace alors une antenne omnidirectionnelle par plusieurs antennes directionnelles, ce qui limite le risque de collision aux messages en provenance d'une même direction. Le facteur de réduction n'est pas exactement égal au nombre d'antennes, car les lobes de réception de chaque antenne se superposent en partie. On obtient donc finalement :

$$\begin{aligned} P_{réception} &= \prod_j p_j(0) \\ &= \prod_j \exp\Big(-\frac{\lambda_j}{\gamma} * (T_{ADS-B} + T_j)\Big) \\ &= \exp\Big(\sum_j -\frac{\lambda_j}{\gamma} * (T_{ADS-B} + T_j)\Big) \end{aligned} \tag{5.6}$$

Cependant, la RTCA-DO-242 ne requiert pas une probabilité de réception minimale pour chaque message, mais plutôt d'avoir une actualisation de la position de l'avion au moins une fois par période de mise à jour. Si le message de position est émis n fois par période de mise à jour, la probabilité finale sera donc :

$$\begin{aligned} P_{actualisation} &= 1 - [1 - P_{réception}]^n \\ &= 1 - \Big[1 - \exp\Big(\sum_j -\frac{\lambda_j}{\gamma} * (T_{ADS-B} + T_j)\Big)\Big]^n \end{aligned} \tag{5.7}$$

5.6.3 Types de messages à considérer

Nous venons de voir que la probabilité de mise à jour réussie dépend de tous les types de messages utilisant la fréquence de 1090 MHz. Ceux-ci sont nombreux, et il n'est pas toujours évident de connaître ni leur taille ni leur fréquence exacte d'émission.

Il y a tout d'abord les messages mode S de réponse aux interrogations des radars secondaires, qui peuvent avoir une longueur de 56 ou 112 bits. La fréquence d'interrogation peut varier d'un radar à l'autre, mais est toujours inférieure à celle d'émission de l'ADS-B. L'utilisation de bande passante en réponse aux SSR sera donc également inférieure à celle de l'ADS-B. De plus, les SSR de dernière génération n'envoient plus des requêtes généralisées à tous les avions, mais seulement à ceux dont ils ont besoin d'une mise à jour. Puisque la position d'un avion émettant de l'ADS-B est déjà connue, il ne recevra pas de requêtes. En fin de compte, nous pouvons donc considérer que tous les avions sont équipés d'ADS-B. Le résultat que nous obtiendrons sera alors une borne inférieure de la probabilité réelle d'actualisation.

Le TCAS, c'est-à-dire le système d'évitement des collisions, utilise également des messages mode S sur la bande de fréquences de 1090 MHz. L'influence du TCAS sur la probabilité d'actualisation de la position d'un avion a justement été étudiée par Rose et al. (2010), toujours du laboratoire *Lincoln*. La spécification du TCAS requiert que celui-ci n'utilise pas

plus de 2% de la capacité de réception d'un transpondeur. Dans le cas où de trop nombreux risques de collision sont détectés, entraînant une utilisation de la bande passante dépassant les 2%, la puissance d'émission des messages TCAS est automatiquement réduite. Finalement, les chercheurs concluent que les mesures de limitation des interférences mises en place pour le TCAS permettent d'en réduire la surcharge au niveau du transpondeur à une moyenne de 2 %, et que le chiffre de 3.5 % n'est presque jamais dépassé. Pour nos simulations, nous augmenterons donc de 3.5 % l'utilisation de la bande passante par rapport aux seuls messages SAT et ADS-B pour prendre en compte le TCAS.

Enfin, la fréquence de 1090 MHz est également utilisée pour certains protocoles de communication militaire, dont les caractéristiques exactes ne sont pas dévoilées. Cependant, leur utilisation de la bande de fréquence est négligeable par rapport à celle des communications civiles. Nous nous contenterons donc de l'ignorer.

5.6.4 Résultats

Nous souhaitons calculer la probabilité de mise à jour réussie de la position d'un avion en fonction de la durée de l'intervalle de la chaîne de clés TESLA pour chacune des trois situations de vol décrites au tableau 5.2. Au regard de toutes les considérations antérieures, les types de messages que nous prenons en compte pour le calcul des probabilités de collision de l'équation 5.7 sont les suivants :

Les autres messages ADS-B émis en moyenne 4.2 fois par seconde (deux fois par seconde pour l'altitude et la position et une fois toutes les cinq secondes pour l'indicatif de vol, comme décrit en 2.2). Ils ont une durée de 144 μs - 120 μs pour le message ADS-B classique, plus 16 μs pour le hash et 8 μs pour le numéro de séquence (un bit représente un temps d'émission de 1 μs puisque le débit binaire est de 1 $Mbit/s$)

Les messages de diffusion d'une clé antérieure émis une fois par intervalle. Ils ont une durée de 192 μs. Celle-ci est due aux 128 bits de la clé en elle-même et aux 8 μs du préambule, aux 5 bits pour le format du message, aux 3 bits pour les capacités du transpondeur, aux 24 bits de l'identifiant unique de l'avion et aux 24 bits de CRC d'un message ADS-B classique.

Les messages de diffusion du certificat de l'avion et de signature de sa chaîne de clés émis une fois toutes les 30 secondes. Leur durée est de 1528 μs. Le certificat est composé de 24 bits pour l'identifiant de l'avion, 32 bits pour la date d'expiration, 256 bits pour la clé publique de l'avion et 512 bits pour la signature ECDSA. Il faut y ajouter la

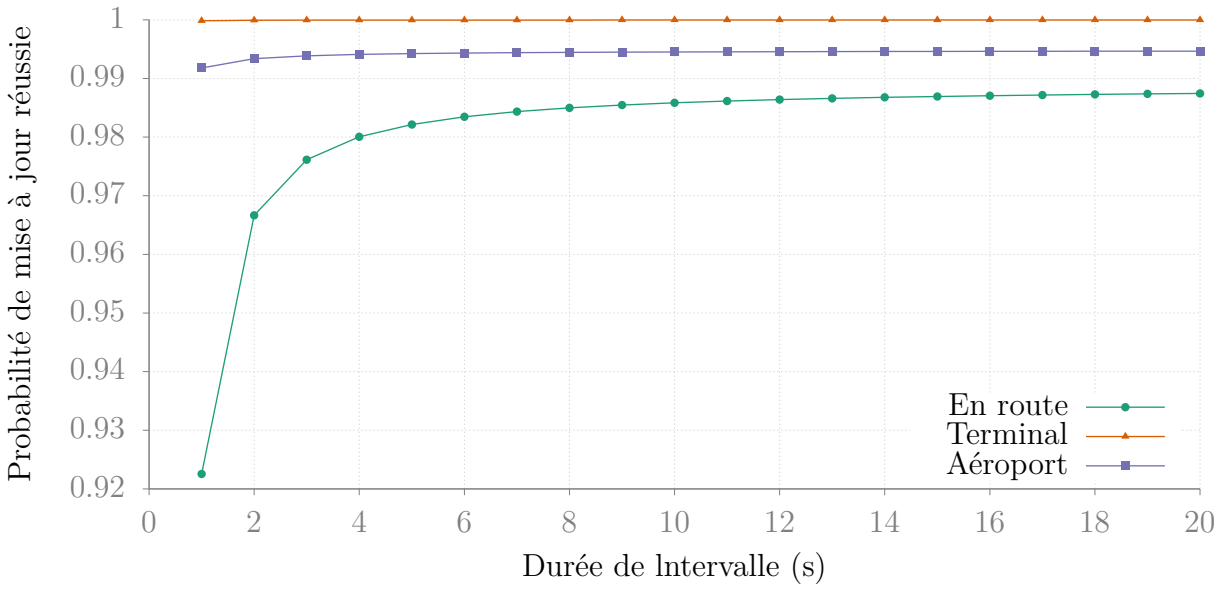


Figure 5.2 Probabilités de mise à jour réussie en fonction de la durée de l'intervalle selon les critères du tableau 5.2

dernière clé publiée de la chaîne de l'avion d'une taille de 128 bits et la signature ECDSA correspondante produite avec la clé publique de l'avion, de 512 bits. Enfin, les éléments d'un message ADS-B classique rajoutent 64 μs.

Les messages TCAS pris en compte en multipliant par 3.5 % l'utilisation de la bande passante :

$$P_{réception} = \exp\left(\left(1 + \frac{3.5}{100}\right) * \sum_{j} -\frac{\lambda_j}{\gamma} * (T_{ADS-B} + T_j)\right) \tag{5.8}$$

La figure 5.2 présente la probabilité que la mise à jour des coordonnées de l'avion ait réussi pour chacune des situations de vol du tableau 5.2, en fonction de la durée de l'intervalle. On remarque que dans les cas *Terminal* et *Aéroport*, la probabilité reste toujours au-dessus des 98 % requis, quelle que soit la durée de l'intervalle. En revanche, pour le cas *En route*, une durée de l'intervalle inférieure à 4 secondes ne satisfait pas la contrainte.

Nous choisissons finalement une durée de 5 secondes pour l'intervalle de la chaîne de clés TESLA, ce qui correspond à un délai identique avant l'authentification de chaque message *jaune*. Il est préférable de garder une certaine marge par rapport à la valeur minimale de 4 seconde. En effet, il se peut que parfois certains types de messages autres que ceux de

position, d'altitude et d'authentification soient émis.

La valeur de 5 secondes pour le délai d'authentification peut être considérée comme satisfaisante. En effet, auparavant les radars ne permettaient aux contrôleurs de n'obtenir une mise à jour de la position d'un avion que toutes les 5 à 12 secondes (OACI Asie et Pacifique, 2008). SAT fait donc aussi bien que les meilleurs radars, et même mieux, car dans notre cas les mises à jour de la position se font à chaque réception d'un message ADS-B, deux fois par seconde. Ce sont certes des messages *jaunes* qui ne deviendront *verts* qu'au bout d'un maximum de 5 secondes, mais ils sont tout de même utiles immédiatement avec un fort degré de fiabilité comme expliqué en 4.4.

Enfin, les calculs que nous avons effectués sont pessimistes. En effet, ils considèrent que la superposition de ne serait-ce que d'un bit de deux messages entraîne la perte de chacun d'entre eux. Or, la modulation PPM utilisée fait en sorte que celui des deux messages dont le signal reçu a la plus forte puissance a une grande probabilité de pouvoir être récupéré. De plus, le code de redondance cyclique peut permettre de corriger un certain nombre d'erreurs du message, comme expliqué par Gertz (1984). La probabilité de 98 % mise à jour réussie sera donc largement respectée.

5.7 Production de la chaîne de clés

Il reste enfin les paramètres de production de la chaîne de clés TESLA à définir. Nous avons vu qu'il était préférable de n'utiliser qu'une unique chaîne de clés pour un vol donné, car autrement il ne serait plus possible d'authentifier des paquets d'une chaîne antérieure du vol.

Nous devons choisir une heure de début de la chaîne commune à tous les avions, de telle façon qu'ils se trouvent à tout instant dans le même intervalle. Cela permet une synchronisation implicite entre avions en utilisant le temps GNSS, et économise la bande passante qui serait autrement utilisée par le processus de synchronisation. Le choix de l'heure de début de chaîne a peu d'importance pratique, et nous décidons arbitrairement de la fixer à 00:00 UTC (Coordinated Universal Time) - il est important d'utiliser une base de temps reconnue internationalement.

Nous choisissons de produire des chaînes d'une durée de 48 heures. En effet, l'autonomie maximum de vol actuelle pour les avions les plus performants est d'une vingtaine d'heures. Cependant, il se peut qu'un avion décolle le soir et atterrisse le lendemain. Puisque la chaîne de clés a débuté la veille à minuit, une durée de 24 heures serait alors dépassée. Le choix d'une durée de 48 heures permet au pire des cas une durée de vol de 24 heures. Cependant, cela entraîne un léger problème. Les chaînes de tous les avions ne débuteront plus forcément

le même jour, puisque certaines auront commencé la veille. Pour contourner ce problème, il suffit de numéroter les jours alternativement *pairs* et *impairs*. Au moment de l'émission du certificat de l'avion et de la signature de la chaîne de clés, un bit sera rajouté pour identifier le jour de début de la chaîne, ce qui représente une utilisation négligeable de la bande passante.

Finalement, une durée de la chaîne de 48 heures avec des intervalles de 5 secondes représente une taille de chaîne de 34560 clés. Avec des clés de 128 bits, cela représente une capacité de stockage de 540 Koctets.

5.8 Résumé des paramètres choisis

Les figures 5.3 et 5.4 résument respectivement les tailles des différents éléments ajoutés par SAT à l'ADS-B classique et les valeurs des autres paramètres que nous avons dû définir.

Tableau 5.3 Récapitulatif de la taille des différents éléments de SAT

	Taille (bits)
Clé publique de l'avion	256
Signature ECDSA	512
Clés TESLA	128
MAC	16
Numéro de séquence	8

Tableau 5.4 Valeur des autres paramètres de SAT

	Valeur
Période d'émission des certificats	30 s
Durée de l'intervalle TESLA	5 s
Début de la chaîne de clés	00:00 UTC
Durée de la chaîne	48 heures
Nombre de clés de la chaîne	34560
Taille de la chaîne	540 Koctets

5.9 Comparaison des résultats de SAT

La comparaison de la probabilité de réception d'un message entre la version actuelle de l'ADS-B et sa version sécurisée SAT est présentée à la figure 5.3. Celle-ci présente également l'évolution de la probabilité en fonction du nombre d'avions présents dans le domaine d'opération. Nous remarquons que la différence entre les deux augmente avec le nombre d'avions présents. Ceci est normal, car si plus d'avions sont présents, la bande passante est plus saturée et donc une augmentation du nombre de messages et de leur taille sera plus dommageable. Nous pouvons toutefois noter que la probabilité de réception ne chute au maximum que de 10 % avec SAT, pour le nombre maximum d'avions requis par la RTCA-DO-242.

Nous pouvons également comparer le surcoût de la sécurisation de l'ADS-B entre SAT et un protocole qui utiliserait de la cryptographie asymétrique traditionnelle à base de signatures pour chaque message. Dans le cas de SAT, nous avons vu en 5.6.4 qu'il fallait rajouter aux messages ADS-B classiques :

1. 24 bits par message pour le MAC et le numéro de séquence.
2. 192 bits toutes les 5 secondes pour la diffusion des clés TESLA. Si ce chiffre est rapporté aux 4.2 messages émis en moyenne chaque seconde, cela représente un équivalent de 9 bits par message.
3. 1528 bits toutes les 30 secondes pour les certificats. Si l'on répartit cela sur tous les messages ADS-B émis sur la période, cela représente une moyenne de 12 bits par message.

Nous obtenons finalement un surcoût équivalent à 45 bits par message, soit une augmentation de 40% par rapport aux messages ADS-B non sécurisés.

Si nous avions choisi de simplement ajouter une signature ECDSA à la fin de chaque paquet ADS-B, cela aurait représenté un supplément de 512 bits par message. C'est une augmentation de 457 % de la taille du message, ce qui est très supérieur à ce que nous obtenons avec SAT.

Nous pourrions également penser répartir la signature sur tous les messages envoyés pendant un intervalle de 5 secondes. Ce serait l'équivalent du déplacement du tampon vers l'émetteur comme discuté en 3.2 pour TESLA. Au bout de 5 secondes, nous appliquerions une signature sur l'ensemble des messages stockés dans le tampon pendant la durée de 5 secondes, et répartirions cette signature sur l'ensemble des messages. Le surcoût serait alors uniquement de 24 bits par message. Cependant, le délai introduit par cette méthode correspond à un délai pour la diffusion des messages et non pas uniquement pour leur authentification comme c'est le cas avec SAT. Les données ne sont plus envoyées en temps réel, et la qualité de service de

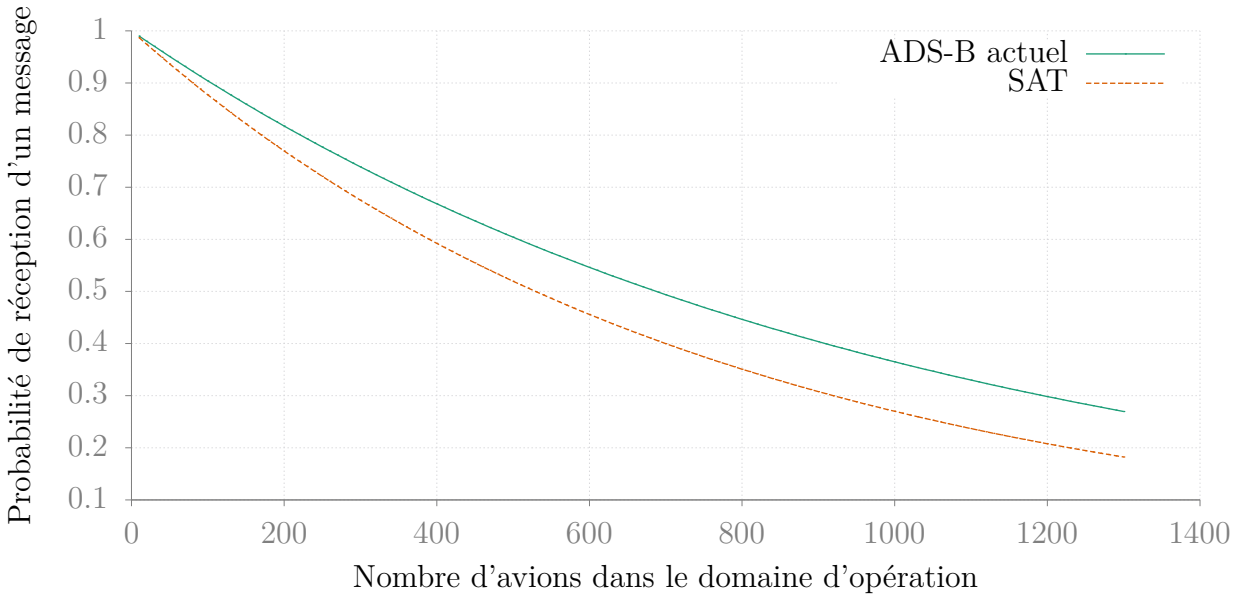


Figure 5.3 Probabilité du succès de la réception d'un message en fonction du nombre d'avions dans le domaine d'opération

la version actuelle de l'ADS-B est réduite. Cela est contraire à nos objectifs et n'est donc pas convenable.

En conclusion, les résultats obtenus par SAT en matière de qualité de service et de surcoût d'utilisation de la bande passante sont bien supérieurs à ceux de la cryptographie asymétrique traditionnelle. La relative complexité introduite par TESLA s'avère donc justifiée.

CHAPITRE 6 IMPLÉMENTATION DE SAT

Parmi nos questions de recherche se trouvaient la facilité d'adoption de SAT, et son niveau de compatibilité avec les transpondeurs actuels. Comme nous avons désormais étudié les détails techniques de notre protocole, nous allons maintenant pouvoir traiter ces interrogations. Nous ne pourrons cependant y répondre qu'en partie. En effet, le processus de tests et de certification est très long, coûteux et complexe en aéronautique.

6.1 Procédures de test avant l'obtention de la certification de vol

Une première étape - que nous avons effectuée et qui sera détaillée au cours de ce chapitre - consiste à implémenter SAT sur radio logicielle (SDR, Software Defined Radio). Cela requiert des moyens limités, puisque nous n'avons pas besoin d'avoir accès à du matériel certifié, qui coûte très cher. La transmission entre l'émetteur et le récepteur doit cependant se faire de façon filaire afin de ne pas perturber le trafic aérien en envoyant de faux messages dans les airs.

Une seconde étape pourrait consister à se placer à l'intérieur d'une cage de Faraday afin de bloquer toute diffusion de nos messages ADS-B en dehors de la zone d'essais. Nous pourrions alors acheter un transpondeur certifié et émettre par ondes radio des messages à partir du SDR implémentant SAT. Si le transpondeur s'avérait capable de recevoir les messages correctement, nous pourrions alors être tentés de conclure que la rétrocompatibilité de notre protocole est confirmée.

Cependant, pour que SAT soit certifié en vol, une batterie de tests supplémentaires devrait être effectuée. Il faudrait acheter un banc d'essai mode S certifié, et appliquer à notre transpondeur une série de tests très précis (EUROCONTROL, 2001). Ce type d'équipement coûte plusieurs milliers de dollars.

Même après cette série de tests, notre transpondeur ne serait toujours pas certifié pour équiper des avions. Nous serions seulement autorisés à poursuivre des essais en vol. Si ces essais sont validés par la NAA - par exemple la FAA aux États-Unis - et que le transpondeur passe également tous les autres tests nécessaires pour s'assurer de sa conformité à des normes strictes - les risques de panne en cours de vol doivent être très faibles -, alors seulement le transpondeur SAT pourra être commercialisé.

6.2 Implémentation sur radio logicielle

Au moment de l'écriture de ce mémoire, le temps et les moyens à notre disposition ne nous ont permis d'effectuer que la première étape de la série de tests décrite précédemment, à savoir l'implémentation sur radio logicielle. L'ensemble du travail a été effectué à l'aide de deux USRP B200 de la société *Ettus Research*, et le code est disponible sur *github* à la fois pour l'émission [1] et la réception [2].

6.2.1 Réception de messages ADS-B standards

Dans un premier temps, nous avons souhaité nous servir de l'une de nos deux radios logicielles pour implémenter simplement la réception de messages ADS-B classiques. Nous avons fait ce choix non seulement pour nous permettre de mieux appréhender le fonctionnement des USRP, mais également pour avoir à notre disposition un catalogue de messages ADS-B valides en provenance d'avions réels.

La programmation des USRP se fait par l'intermédiaire du logiciel libre *gnuradio*. L'ADS-B utilise la modulation PPM - comme décrit au chapitre 2 -, qui est relativement simple. Cependant, l'encodage des données de position est assez complexe, car elle utilise un format spécifique - nommé Compact Position Reporting (CPR) - afin d'économiser la quantité de données nécessaires. De même, chaque type de message a ses spécificités d'encodage, et il aurait été long et laborieux d'implémenter le décodage de chacun d'entre eux.

Or, il se trouve qu'un logiciel a déjà été développé par Foster (2016), permettant la réception de messages mode S - y compris ADS-B - sur SDR (figure 6.1). Il s'agit du module *gr-air-modes* et nous avons donc décidé de l'utiliser. Nous avons branché une antenne 1090 MHz sur notre USRP, et sommes allés en haut du Mont Royal afin d'être dans un environnement dégagé permettant une meilleure réception (figure 2.4). Nous avons collecté des messages sur un intervalle de 10 minutes. Au total, nous en avons obtenu 742 émis par 4 avions distincts.

6.2.2 Émission de messages ADS-B standards

La seconde étape de nos tests a consisté à utiliser notre second USRP afin d'émettre des messages ADS-B standards. Cependant, contrairement à la réception, il n'existe aucun module pour *gnuradio* permettant de le faire. Nous avons donc dû coder le nôtre.

Celui-ci consiste principalement en une modulation PPM à 1 MHz, utilisant le bon préambule

1. `https://github.com/PaulTPT/adsb-send_secure`
2. `https://github.com/PaulTPT/air-modes`

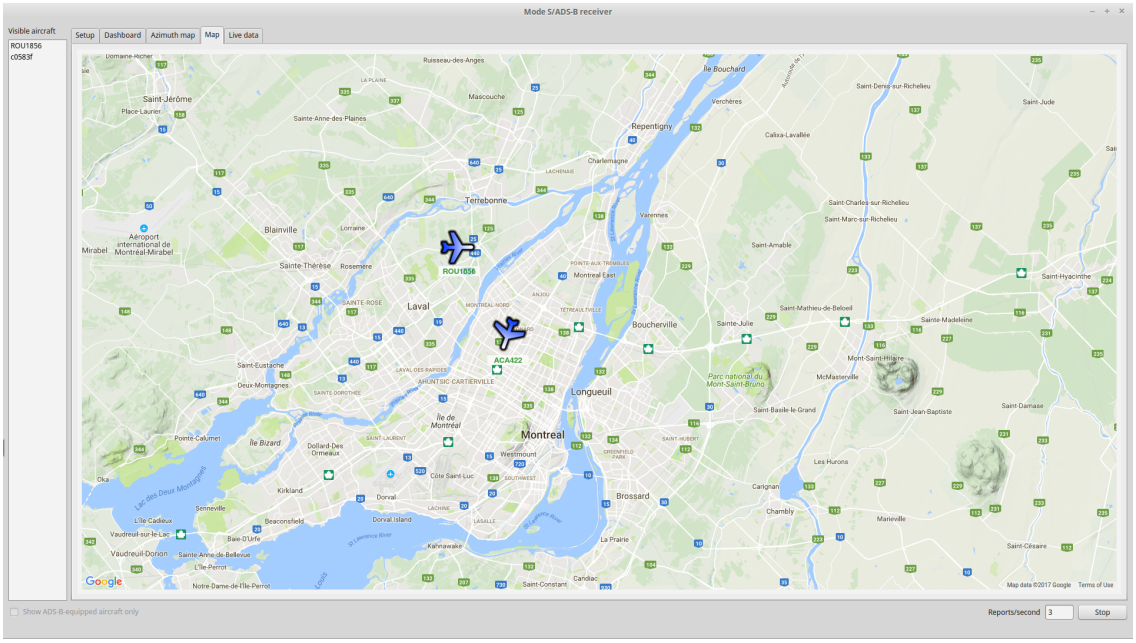


Figure 6.1 Le logiciel *gr-air-modes*. Il reçoit les données mode S en provenance d'une radio logicielle et les décode. Il permet également de visualiser les avions sur une carte à partir de leur position obtenue grâce aux messages ADS-B. Sur cette capture d'écran, on peut observer deux avions d'Air Canada et d'Air Canada Rouge en train d'effectuer leur approche vers l'aéroport Pierre-Elliott-Trudeau de Montréal. Les données ont été capturées depuis le mont Royal.

du mode S. Plutôt que de devoir développer un module de création de messages ADS-B, dont nous n'aurions pas été assurés de la validité, nous avons choisi de réémettre les messages capturés lors de la première étape.

Pour tester si notre implémentation fonctionne correctement, nous avons ensuite retransmis ces messages. Pour ne pas interférer avec le contrôle du trafic aérien réel, nous avons connecté directement nos deux radios logicielles entre elles de façon filaire, et avec un filtre de -5 dB pour ne pas saturer le récepteur. Nous avons alors vérifié que le logiciel *gr-air-modes* était bien capable de décoder à nouveau chacun des messages, et que ceux-ci étaient identiques à ceux de la première capture.

6.2.3 Émission de messages SAT

Nous avons désormais à disposition un émetteur et un récepteur SDR de messages ADS-B standards. À cette étape de nos tests, nous considérons qu'ils sont équivalents respectivement à un transpondeur ADS-B *out* et ADS-B *in*. Nous souhaitons maintenant implémenter une surcouche SAT pour notre émetteur, et vérifier que les messages qu'il produira seront

toujours décodés de façon correcte par le récepteur standard. Afin de gérer les fonctions cryptographiques de production des signatures ECDSA et des HMAC, nous choisissons d'utiliser la bibliothèque *OpenSSL*.

Comme à l'étape précédente, nous connectons désormais l'émetteur SAT et le récepteur ADS-B standard de façon filaire. Nous pouvons alors vérifier que le logiciel *gr-air-modes* est bien capable de décoder chacun des messages, malgré l'ajout du code d'authentification et du numéro de séquence. Ceux-ci ne sont simplement pas pris en compte. Les messages contenant les certificats ou les clés de la chaîne TESLA sont complètement ignorés, car ils utilisent un numéro de format qui n'est pas reconnu.

6.2.4 Réception de messages SAT

La dernière étape consiste à modifier le logiciel *gr-air-modes* afin qu'il soit capable de décoder les nouveaux messages de SAT et de prendre en considération les nouveaux bits ajoutés aux messages ADS-B. Nous utilisons à nouveau la bibliothèque *OpenSSL*, cette fois-ci pour vérifier les signatures ECDSA et reproduire les HMAC afin de les vérifier. Nous devons également implémenter toute la logique de vérification de l'authenticité des messages introduite par SAT, afin d'être capable de classifier ceux-ci dans les quatre catégories *rouge*, *orange*, *jaune* et *verte*.

Nous pouvons désormais effectuer une nouvelle fois les essais de communication entre l'émetteur et le récepteur de façon filaire, chacun d'entre eux implémentant cette fois-ci la surcouche SAT. Nous réémettons tout d'abord les messages réels enregistrés précédemment, comme s'ils étaient produits de façon légitime par notre émetteur. Nous vérifions alors qu'ils sont correctement classifiés comme valides après la divulgation des clés. Nous avons pour cela agi comme autorité de certification et avons fourni le certificat nécessaire à notre USRP. Dans un second temps, nous recommençons l'opération avec les mêmes messages, mais cette fois-ci en remplaçant de temps à autre le MAC authentique par des bits générés de façon aléatoire. Nous vérifions alors que le cas échéant, les messages sont bien classés comme rouges.

6.3 Implémentation sur les transpondeurs actuels

Notre implémentation de SAT et les différents tests que nous avons effectués sur SDR sont concluants, et cela laisse présager que notre protocole permettra aux transpondeurs actuels de correctement continuer à recevoir les messages ADS-B standards. Cependant, comme expliqué en 6.1, cela n'est qu'une première étape et ne prouve pas qu'il en sera de même sur de véritables transpondeurs. Il faudra pour cela poursuivre les différentes étapes de tests.

Un autre point important pour faciliter l'adoption de SAT est la mise à jour des transpondeurs actuels pour qu'ils puissent implémenter le protocole à peu de frais. Cela est nécessaire, car suite à l'obligation d'être équipés de transpondeurs compatibles ADS-B afin d'être autorisés à voler dans une grande partie de l'espace aérien américain, les pilotes sont en train de s'équiper à grands frais. Un renouvellement des transpondeurs dans un futur proche n'est donc pas envisageable.

Notre implémentation de SAT est très semblable à celle de l'ADS-B standard. Nous avons notamment veillé à conserver la modulation et la structure des messages actuels. La principale différence a trait à l'utilisation de fonctions cryptographiques. Il faudra donc s'assurer que les transpondeurs actuels ont les capacités de calcul et de stockage nécessaires. Une fois encore, nos tests laissent présager que ceux-ci pourront facilement évoluer pour être compatible SAT grâce à une simple mise à jour logicielle. Cependant, de nouveaux tests sont nécessaires pour s'en assurer formellement.

CHAPITRE 7 CONCLUSION

L'objectif de ce mémoire était de déterminer l'applicabilité, la performance, et la viabilité du protocole TESLA en ce qui concerne la sécurisation de l'ADS-B. Pour cela, nous devions répondre aux questions suivantes :

Q.1 En quoi les solutions antérieures proposées pour sécuriser l'ADS-B ne sont-elles pas satisfaisantes ?

Q.2 Quelles sont les spécificités du protocole TESLA qui en font une solution intéressante pour la sécurisation de l'ADS-B ?

Q.3 Quelles sont les modifications à apporter à TESLA pour qu'il puisse fonctionner avec l'ADS-B, dans le contexte très spécifique du contrôle du trafic aérien ? Quels sont les points techniques d'implémentation à considérer ?

Q.4 Jusqu'à quel point une augmentation de la quantité de données nécessaires à notre version sécurisée de TESLA est-elle envisageable ? Quel en serait l'impact sur le taux de succès de réception des messages, et donc sur la fiabilité du protocole - critique dans le cadre du transport aérien ?

Q.5 Quelle serait la facilité d'adoption du protocole SAT ? En particulier, quel serait le niveau de compatibilité avec les transpondeurs actuels ?

Nous allons désormais synthétiser nos travaux pour montrer comment nous avons répondu à ces différentes questions à travers notre nouveau protocole, SAT. Nous décrirons ensuite les limites de notre solution, ainsi que les travaux et améliorations futures qui pourraient corriger ses quelques défauts.

7.1 Synthèse des travaux

Q.1 Après avoir étudié les caractéristiques du contrôle du trafic aérien et ses contraintes, nous avons expliqué que les principaux objectifs de sécurité à atteindre étaient l'authentification et la vérification de l'intégrité des messages émis par les avions. Nous nous sommes ensuite rendu compte que les travaux antérieurs ne répondaient pas à ces objectifs de manière satisfaisante. La vérification de la validité des données reçues - par exemple par multilatération - peut fonctionner dans certains cas, mais n'est pas suffisamment fiable. Les méthodes cryptographiques traditionnelles, quant à elles, ne sont pas applicables. La cryptographie symétrique ne convient pas à la diffusion *broadcast* de l'ADS-B. En effet, cela supposerait de

diffuser largement les clés utilisées, ce qui induirait un risque élevé qu'elles tombent entre les mains d'un attaquant. Enfin, la cryptographie asymétrique et ses signatures utilisent trop largement la bande passante, et ne sont donc pas non plus applicables à grande échelle.

Q.2 Puisqu'aucune méthode antérieure ne nous semblait réellement applicable, nous avons présenté une nouvelle solution potentielle, le protocole TESLA. Celui-ci cumule les avantages de la cryptographie asymétrique - à savoir la non-divulgation de clés valides pour l'authentification de messages - et ceux de la cryptographie symétrique - c'est-à-dire de ne consommer que peu de bande passante grâce à l'utilisation de courts codes d'authentification plutôt que de signatures.

Q.3 Nous avons ensuite introduit nos propres modifications à TESLA afin de concevoir le protocole SAT, adapté spécifiquement au contexte du contrôle du trafic aérien. Celui-ci remplace notamment le mécanisme de synchronisation temporelle initial de TESLA par l'utilisation du temps GNSS. Une infrastructure à clés publiques basée sur la structure organisationnelle de l'ATC - tant au niveau mondial que national - a également été mise en place.

Le principal inconvénient de TESLA, et par extension de SAT, est l'introduction d'un délai pour l'authentification des messages - bien que ces derniers soient toujours transmis en temps réel. Nous avons donc défini différents niveaux de confiance pour les messages, selon l'étape du processus d'authentification dans laquelle ils sont rendus. Cela permet tant aux pilotes qu'aux contrôleurs de les prendre en considération immédiatement à leur réception, en suivant des procédures particulières selon leur code de couleur. La problématique du délai est ainsi diminuée.

Q.4 On peut aussi agir directement sur le délai d'authentification en le réduisant au maximum, mais cela entraîne une augmentation de l'utilisation de la bande passante. Cette dernière dépend également des tailles des clés, des codes d'authentification, et d'autres données introduites par SAT. Nous avons donc dû discuter du meilleur compromis entre tous ces différents paramètres, et de l'impact correspondant sur le taux de succès de réception des messages. Nous sommes parvenus à la conclusion qu'un délai de 5 secondes restait convenable tant pour les pilotes que pour les contrôleurs. Cette valeur, associée à des tailles convenables pour les autres paramètres - la sécurité de SAT au-delà de 2030 est assurée - permet de respecter toutes les contraintes de la spécification de l'ADS-B. En particulier, SAT sera capable d'absorber la croissance constante du volume du trafic aérien lors des prochaines années.

Q.5 Nous avons enfin amorcé le processus d'implémentation et de tests de notre protocole afin de nous assurer qu'il serait compatible avec les transpondeurs actuels. En effet, l'introduction de l'ADS-B a déjà forcé leur remplacement aux pilotes, et ce processus ne pourra pas être renouvelé à court ou moyen terme, car il est très coûteux. De plus, la mise en place d'un nouveau protocole requiert un certain temps, et SAT se devait donc d'être rétrocompatible avec l'ADS-B afin que le système d'ATC reste fonctionnel tout au long de la transition. Nos premiers essais sur radio logicielle sont concluants et laissent envisager que le critère de rétrocompatibilité sera rempli, mais de nombreuses étapes de tests restent indispensables avant d'obtenir les certifications nécessaires à un déploiement en vol de SAT.

7.2 Limitations de la solution proposée

Des limitations avaient été envisagées pour TESLA dans la littérature antérieure, alors même qu'aucune véritable étude de l'applicabilité du protocole à la sécurisation de l'ADS-B n'avait été effectuée.

Ainsi, Yang et al. (2014) expliquaient que l'utilisation de TESLA n'était pas envisageable dans le contexte du contrôle du trafic aérien à cause du délai d'authentification. Nous avons finalement démontré le contraire.

Pour sa part, Cook (2015) considérait que l'utilisation de TESLA n'était pas possible, car il faudrait connaître à l'avance la durée du vol afin d'être capable de générer la chaîne de clés. Nous avons choisi d'utiliser une chaîne de taille fixe, ce qui contredit cet argument. De plus, le stockage des messages dans un tampon avant leur authentification saturerait la mémoire du transpondeur. Or, 192 bits par message, émis en moyenne 4.2 fois par seconde pour une durée de 5 secondes et pour au maximum 1250 avions (comme détaillé en 5.6) ne représentent que 615 Kio. Cela reste suffisamment faible en comparaison des tailles de mémoire désormais disponibles.

En revanche, la véritable limitation de SAT est l'augmentation de 40% de l'utilisation de bande passante par rapport à l'ADS-B traditionnel. Si ce résultat nous permet de rester dans les normes définies par la spécification du protocole, il empêche toutefois d'autant l'utilisation des messages complémentaires de l'ADS-B, comme les caractéristiques techniques de l'avion ou les prochaines étapes du plan de vol. Ces informations ne sont toutefois pas indispensables, et leur utilisation demeure très rare.

Le léger délai de cinq secondes à l'authentification, s'il reste acceptable selon notre discussion avec le panel de pilotes, reste enfin à être éprouvé lors de tests en vol, pour s'assurer qu'il n'est pas trop important en pratique.

7.3 Améliorations futures

Si le protocole SAT répond d'ores et déjà aux spécifications de l'ADS-B, un certain nombre de travaux restent à effectuer avant qu'il ne puisse obtenir sa certification et être utilisé internationalement en remplacement de l'ADS-B.

Comme décrit en 4.2.3.3, l'utilisation de signatures basées sur l'identité pourrait être un moyen d'économiser encore de la bande passante en retirant la nécessité de transmettre le certificat de l'avion. Cependant, tous les détails d'implémentation restent à définir.

Il faudra également continuer à travailler avec des panels de pilotes et de contrôleurs afin de déterminer la façon optimale de leur présenter les informations SAT. Ils doivent en effet pouvoir facilement et rapidement comprendre quels sont les messages dignes de confiance. La classification de ces derniers par couleur est un grand pas dans cette direction, mais nous devons encore déterminer en détail comment seront représentés les avions et leurs trajectoires. Les procédures qu'il faudrait mettre en place en cas de doutes sur la légitimité d'un message sont également à déterminer.

Il faudra bien sûr ensuite continuer les différentes étapes de test décrites en 6.1, avant d'espérer obtenir la certification pour un transpondeur SAT. Il faudra également convaincre l'OACI des qualités de notre solution, pour qu'elle soit reconnue comme un standard international en remplacement de l'ADS-B.

Enfin, si SAT a été pensé spécifiquement pour s'adapter de façon optimale au contrôle du trafic aérien des avions, le protocole peut-il s'avérer performant dans d'autres situations ? Avec le fort développement des drones au cours de ces dernières années - qui devrait continuer à croître exponentiellement dans le futur -, l'utilisation de SAT pour cette catégorie d'aéronefs est à discuter. La bande passante disponible sera-t-elle toujours suffisante ? Comment un drone sera-t-il capable de comprendre et réagir aux différentes catégories de messages ? Voici autant de questions qui demandent réflexion et qui pourraient orienter les travaux futurs dans ce domaine.

RÉFÉRENCES

Airbus, “Airbus Global market forecast 2016-2035”, Airbus, Rapp. tech., 2016.

Aireon, “Space-Based ADS-B Global Air Traffic Surveillance”, 2016. En ligne : http://aireon.com/

S. Amin, T. Clark, R. Offutt, et K. Serenko, “Design of a cyber security framework for ADS-B based surveillance systems”, dans Systems and Information Engineering Design Symposium (SIEDS), 2014. IEEE, 2014, pp. 304–309. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6829910

ANSSI, “Mécanismes cryptographiques - Règles et recommandations concernant le choix et le dimmensionnement des mécanismes cryptographiques”, Fév. 2014. En ligne : http://www.ssi.gouv.fr/uploads/2015/01/RGS_v-2-0_B1.pdf

APANPIRG/26, “ADS-B implementation and operations guidance document”, ICAO Asia and Pacific, Rapp. tech., Sep. 2015.

J. Baek, Y.-J. Byon, E. Hableel, et M. Al-Qutayri, “An Authentication Framework for Automatic Dependent Surveillance-Broadcast Based on Online/Offline Identity-Based Signature”, dans 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing. IEEE, Oct. 2013, pp. 358–363. DOI : 10.1109/3PGCIC.2013.61. En ligne : http://ieeexplore.ieee.org/document/6681254/

E. Barker, “Recommendation for Key Management Part 1 : General”, National Institute of Standards and Technology, Rapp. tech. NIST SP 800-57pt1r4, Jan. 2016, dOI : 10.6028/NIST.SP.800-57pt1r4. En ligne : http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf

G. T. Becker, S. Lo, D. De Lorenzo, D. Qiu, C. Paar, et P. Enge, “Efficient authentication mechanisms for navigation systems–a radio-navigation case study”, dans Proceedings of the ION GNSS Meeting, 2009. En ligne : https://www.ei.rub.de/media/crypto/veroeffentlichungen/2010/09/05/beckeriongnss09.pdf

M. Bellare, R. Canetti, et H. Krawczyk, “Keying Hash Functions for Message Authentication”, dans Advances in Cryptology — CRYPTO ’96. Springer, Berlin,

Heidelberg, Août 1996, pp. 1–15, dOI : 10.1007/3-540-68697-5_1. En ligne : http://link.springer.com/chapter/10.1007/3-540-68697-5_1

W. Blacker, “FAA - Aircraft Situation Display to Industry”, 2013. En ligne : http://www.fly.faa.gov/ASDI/asdi.html

R. E. Boisvert et V. A. Orlando, “ADS-Mode S system overview”, dans Digital Avionics Systems Conference, 1993. 12th DASC., AIAA/IEEE. IEEE, 1993, pp. 104–109. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=283562

E. Cook, “ADS-B, Friend or Foe : ADS-B Message Authentication for NextGen Aircraft”, dans 7th International Symposium on Cyberspace Safety and Security (CSS). IEEE, Août 2015, pp. 1256–1261. DOI : 10.1109/HPCC-CSS-ICESS.2015.201. En ligne : http://ieeexplore.ieee.org/document/7336340/

A. Costin et A. Francillon, “Ghost in the Air (Traffic) : On insecurity of ADS-B protocol and practical attacks on ADS-B devices”, Black Hat USA, pp. 1–12, 2012. En ligne : https://www.radarspotters.eu/software/BH_US_12_Costin_Ghosts_In_Air_WP.pdf

ECRYPT II, “Yearly Report on Algorithms and Keysizes”, Sep. 2012. En ligne : http://link.springer.com/chapter/10.1007/11586821_2

M. H. Eldefrawy, M. K. Khan, K. Alghathbar, et E.-S. Cho, “Broadcast Authentication for Wireless Sensor Networks Using Nested Hashing and the Chinese Remainder Theorem”, Sensors, vol. 10, no. 9, pp. 8683–8695, Sep. 2010. DOI : 10.3390/s100908683. En ligne : http://www.mdpi.com/1424-8220/10/9/8683

EUROCONTROL, “Mode S transponders test benches functional requirements”, Fév. 2001. En ligne : http://www.eurocontrol.int/sites/default/files/library/005_Modes_S_Transponder_Test_Benches_Requirements.pdf

C. Finke, J. Butts, R. Mills, et M. Grimaila, “Enhancing the security of aircraft surveillance in the next generation air traffic control system”, International Journal of Critical Infrastructure Protection, vol. 6, no. 1, pp. 3–11, Mars 2013. DOI : 10.1016/j.ijcip.2013.02.001. En ligne : http://linkinghub.elsevier.com/retrieve/pii/S1874548213000048

FlightAware, “Flight Tracker”, 2016. En ligne : http://flightaware.com/

FlightRadar, “Flightradar24.com - Live flight tracker!” 2016. En ligne : https://www.flightradar24.com/46.69,-76.41/5

N. Foster, "'gr-air-modes', a Mode S/ADS-B receiver for the Gnuradio software-defined radio project", 2016. En ligne : github.com/bistromath/gr-air-modes

G. Galati, M. Leonardi, P. Magaro, et V. Paciucci, "Wide area surveillance using SSR mode S multilateration : advantages and limitations", dans European Radar Conference, 2005. EURAD 2005., Oct. 2005, pp. 225–229. DOI : 10.1109/EURAD.2005.1605606

R. Gennaro et P. Rohatgi, "How to sign digital streams", dans Advances in Cryptology — CRYPTO '97. Springer, Berlin, Heidelberg, Août 1997, pp. 180–197, dOI : 10.1007/BFb0052235. En ligne : https://link.springer.com/chapter/10.1007/BFb0052235

J. L. Gertz, "Fundamentals of Mode S Parity Coding", MIT Lincoln laboratory, Rapp. tech., Avr. 1984.

D. Giry et J.-J. Quisquater, "BlueKrypt Cryptographic key length recommendation", Sep. 2015. En ligne : https://www.keylength.com/en/compare/

GPS directorate, "IS-GPS-200h : Global Positionning Systems directorate, systems engineering & integration. Interface specification." Sep. 2013.

B. Haines, "Hackers + Airplanes No Good Can Come Of This", Las Vegas, Juil. 2012. En ligne : https://www.defcon.org/images/defcon-20/dc-20-presentations/Renderman/DEFCON-20-RenderMan-Hackers-plus-Airplanes.pdf

Y. Haomiao, K. Hyunsung, L. Hongwei, Y. Eunjun, W. Xiaofen, et D. Xuefeng, "An Efficient Broadcast Authentication Scheme with Batch Verification for ADS-B Messages", KSII Transactions on Internet and Information Systems, vol. 7, no. 10, pp. 2544–2560, Oct. 2013. DOI : 10.3837/tiis.2013.10.013. En ligne : http://ksii.cafe24.com/download.jsp?filename=TIISVol7,No10-13.pdf

C. J. Hegarty et E. Chatre, "Evolution of the Global Navigation SatelliteSystem (GNSS)", Proceedings of the IEEE, vol. 96, no. 12, pp. 1902–1917, Déc. 2008. DOI : 10.1109/JPROC.2008.2006090

T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, et P. M. Kintner Jr, "Assessing the spoofing threat : Development of a portable GPS civilian spoofer", dans Proceedings of the ION GNSS international technical meeting of the satellite division, vol. 55, 2008, p. 56. En ligne : https://gps.mae.cornell.edu/humphreys_etal_iongnss2008.pdf

T. Kacem, D. Wijesekera, et P. Costa, “Integrity and authenticity of ADS-B broadcasts”, dans 2015 IEEE Aerospace Conference. IEEE, 2015, pp. 1–8. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7119293

T. Kacem, D. Wijesekera, P. Costa, J. Carvalho, M. Monteiro, et A. Barreto, “Key distribution mechanism in secure ADS-B networks”, dans 2015 Integrated Communication, Navigation and Surveillance Conference (ICNS). IEEE, 2015, pp. P3–1. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7121252

A. R. Karmarkar et L. Martin, “Aviation communication infrastructure security”, dans Integrated Communications, Navigation and Surveillance Conference (ICNS), 2012. IEEE, 2012, pp. E7–1. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6218392

R. Kaune, C. Steffes, S. Rau, W. Konle, et J. Pagel, “Wide area multilateration using ADS-B transponder signals”, dans 2012 15th International Conference on Information Fusion, Juil. 2012, pp. 727–734.

B. Kovell, B. Mellish, T. Newman, et O. Kajopaiye, “Comparative analysis of ADS-B verification techniques”, The University of Colorado, Boulder, vol. 4, 2012. En ligne : http://morse.colorado.edu/~tlen5710/12s/ADSBVerification.pdf

H. Krawczyk, R. Canetti, et M. Bellare, “RFC 2104 : HMAC : Keyed-Hashing for Message Authentication”, Fév. 1997. En ligne : https://tools.ietf.org/html/rfc2104

J. Krozel, D. Andrisani, M. Ayoubi, T. Hoshizaki, et C. Schwalm, “Aircraft ADS-B Data Integrity Check”, dans AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum. American Institute of Aeronautics and Astronautics, 2004, dOI : 10.2514/6.2004-6263. En ligne : http://arc.aiaa.org/doi/abs/10.2514/6.2004-6263

O. Laffargue, “N’ayez pas peur, l’avion reste le moyen de transport le plus sûr”, Mai 2016. En ligne : http://www.bfmtv.com/international/n-ayez-pas-peur-l-avion-reste-moyen-de-transport-le-plus-sur-975959.html

A. K. Lenstra, “Key lengths”, Wiley, Rapp. tech., 2006. En ligne : https://infoscience.epfl.ch/record/164539/files/NPDF-32.pdf

D. Liu, P. Ning, S. Zhu, et S. Jajodia, “Practical broadcast authentication in sensor networks”, dans The Second Annual International Conference on Mobile and

Ubiquitous Systems : Networking and Services, Juil. 2005, pp. 118–129. DOI : 10.1109/MOBIQUITOUS.2005.49

D. Liu et P. Ning, “Multilevel μTESLA : Broadcast Authentication for Distributed Sensor Networks”, ACM Trans. Embed. Comput. Syst., vol. 3, no. 4, pp. 800–836, Nov. 2004. DOI : 10.1145/1027794.1027800. En ligne : http://doi.acm.org/10.1145/1027794.1027800

S. C. Lo et P. K. Enge, “Authenticating aviation augmentation system broadcasts”, dans IEEE/ION Position, Location and Navigation Symposium, Mai 2010, pp. 708–717. DOI : 10.1109/PLANS.2010.5507223

G. Maxson, “A Brief History of ADS-B”, Déc. 2011. En ligne : http://adsbforgeneralaviation.com/a-brief-history-of-ads-b/

D. McCallie, J. Butts, et R. Mills, “Security analysis of the ADS-B implementation in the next generation air transportation system”, International Journal of Critical Infrastructure Protection, vol. 4, no. 2, pp. 78–87, Août 2011. DOI : 10.1016/j.ijcip.2011.06.001. En ligne : http://linkinghub.elsevier.com/retrieve/pii/S1874548211000229

R. C. Merkle, “A Certified Digital Signature”, dans Advances in Cryptology — CRYPTO’ 89 Proceedings. Springer, New York, NY, Août 1989, pp. 218–238, dOI : 10.1007/0-387-34805-0_21. En ligne : https://link.springer.com/chapter/10.1007/0-387-34805-0_21

NAV CANADA, “Products and services - ADS-B”, 2016. En ligne : http://www.navcanada.ca/EN/products-and-services/Pages/on-board-operational-initiatives-ads-b.aspx

M. S. Nolan, Fundamentals of Air Traffic Control., 5e éd. DELMAR CENGAGE Learning, 2010.

OACI, “Regulations for the ICAO Public Key Directory”, Juil. 2011. En ligne : http://docplayer.net/18227957-Pkd-board-icao-pkd-unclassified-b-tec-36-regulations-for-the-icao-public-key-directory.html

——, “Secondary surveillance radar MODE S advisory circular”, OACI, Montréal, Cirular 174-AN/110, 1983.

OACI, éd., Technical provisions for Mode S Services and extended squitter, 1er éd., série Doc / International Civil Aviation Organization. Montréal : ICAO, 2008, no. 9871, oCLC : 318564865.

OACI Asie et Pacifique, “Guidance material on issues to be considered in ATC multi-sensor fusion processing including the integration of ADS-B data”, APANPIRG/19, Rapp. tech., Sep. 2008.

V. Orlando et W. Harman, “GPS-Squitter Capacity Analysis”, Mai 1994.

W.-J. Pan, Z.-L. Feng, et Y. Wang, “ADS-B data authentication based on ECC and X. 509 certificate”, Journal of Electronic Science and Technology, vol. 10, no. 1, pp. 51–55, 2012. En ligne : http://www.ccs.asia.edu.tw/ezfiles/2/1002/img/374/1201-9.pdf

P. Papadimitratos et A. Jovanovic, “Protection and fundamental vulnerability of GNSS”, dans 2008 IEEE International Workshop on Satellite and Space Communications, Oct. 2008, pp. 167–171. DOI : 10.1109/IWSSC.2008.4656777

A. Perrig, R. Canetti, J. D. Tygar, et D. Song, “Efficient authentication and signing of multicast streams over lossy channels”, dans Proceeding 2000 IEEE Symposium on Security and Privacy. S P 2000, 2000, pp. 56–73. DOI : 10.1109/SECPRI.2000.848446

A. Perrig, J. D. Tygar, B. Briscoe, R. Canetti, et D. Song, “Timed Efficient Stream Loss-Tolerant Authentication (TESLA) : Multicast Source Authentication Transform Introduction”, 2005. En ligne : https://tools.ietf.org/html/rfc4082

A. Perrig, R. Canetti, D. Song, et J. D. Tygar, “Efficient and secure source authentication for multicast”, dans Network and Distributed System Security Symposium, NDSS, vol. 1, 2001, pp. 35–46. En ligne : https://people.eecs.berkeley.edu/~tygar/papers/Efficient_secure_source_authentication.pdf

A. Perrig, R. Canetti, J. D. Tygar, et D. Song, “The TESLA broadcast authentication protocol”, RSA CryptoBytes, vol. 5, 2002. En ligne : http://repository.cmu.edu/epp/62/?utm_source=repository.cmu.edu/epp/62&utm_medium=PDF&utm_campaign=PDFCoverPages

A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, et D. E. Culler, “SPINS : Security Protocols for Sensor Networks”, Wirel. Netw., vol. 8, no. 5, pp. 521–534, Sep. 2002. DOI : 10.1023/A:1016598314198. En ligne : http://dx.doi.org/10.1023/A:1016598314198

L. Purton, H. Abbass, et S. Alam, “Identification of ADS-B system vulnerabilities and threats”, dans Australian Transport Research Forum, Canberra, 2010, pp. 1–16. En ligne : https://www.researchgate.net/profile/Leon_Purton/publication/267957938_Identification_of_ADS-B_System_Vulnerabilities_and_Threats/links/54caa07b0cf22f98631bf519.pdf

R. Robinson, M. Li, S. Lintelman, K. Sampigethaya, R. Poovendran, D. v. Oheimb, et J.-U. BuBer, “Impact of Public Key Enabled Applications on the Operation and Maintenance of Commercial Airplanes”, dans 7th AIAA ATIO Conf, 2nd CEIAT Int’l Conf on Innov and Integr in Aero Sciences,17th LTA Systems Tech Conf; followed by 2nd TEOS Forum. American Institute of Aeronautics and Astronautics, 2007, dOI : 10.2514/6.2007-7769. En ligne : http://arc.aiaa.org/doi/abs/10.2514/6.2007-7769

C. E. Rose, A. D. Panken, W. H. Harman, et L. Wood, “TCAS surveillance performance analysis”, dans Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th. IEEE, 2010, pp. 3–B. En ligne : http://ieeexplore.ieee.org/abstract/document/5655373/

RTCA, “RTCA-DO-242 : Minimum aviation system performance standards for Automatic Dependent Surveillance Broadcast (ADS-B)”, 2002.

RTCA Free flight Select Committee, Safe Flight 21 STeering Committee, et Eurocontro ADS Programme, “Technical Link Assessment Report, APPENDIX F : 1090 MHz Extended Squitter System Description”, Sep. 2001.

K. Sampigethaya, “Visualization & assessment of ADS-B security for green ATM”, dans Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th. IEEE, 2010, pp. 3–A. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5655382

K. Sampigethaya, R. Poovendran, et L. Bushnell, “Assessment and mitigation of cyber exploits in future aircraft surveillance”, dans Aerospace Conference, 2010 IEEE. IEEE, 2010, pp. 1–10. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5446905

L. Schuchman, “Automatic dependent surveillance system secure ADS-S”, U.S. Brevet US7 876 259B2, Jan., 2011. En ligne : http://www.google.com/patents/US7876259B2

M. Schäfer, V. Lenders, et I. Martinovic, “Experimental Analysis of Attacks on Next Generation Air Traffic Communication”, dans Applied Cryptography and Network Security.

Springer, Berlin, Heidelberg, Juin 2013, pp. 253–271, dOI : 10.1007/978-3-642-38980-1_16. En ligne : http://link.springer.com/chapter/10.1007/978-3-642-38980-1_16

A. Smith, R. Cassell, T. Breen, R. Hulstrom, et C. Evers, "Methods to Provide System-Wide ADS-B Back-Up, Validation and Security", dans 2006 ieee/aiaa 25TH Digital Avionics Systems Conference, Oct. 2006, pp. 1–7. DOI : 10.1109/DASC.2006.313681

M. Strohmeier, M. Schafer, V. Lenders, et I. Martinovic, "Realities and challenges of nextgen air traffic management : the case of ADS-B", IEEE Communications Magazine, vol. 52, no. 5, pp. 111–118, 2014. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6815901

M. Strohmeier, V. Lenders, et I. Martinovic, "Lightweight Location Verification in Air Traffic Surveillance Networks", dans Proceedings of the 1st ACM Workshop on Cyber-Physical System Security, série CPSS '15. New York, NY, USA : ACM, 2015, pp. 49–60. DOI : 10.1145/2732198.2732202. En ligne : http://doi.acm.org/10.1145/2732198.2732202

——, "On the Security of the Automatic Dependent Surveillance-Broadcast Protocol", IEEE Communications Surveys & Tutorials, vol. 17, no. 2, pp. 1066–1087, 2015. DOI : 10.1109/COMST.2014.2365951. En ligne : http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6940209

J. Taylor, "Global ATC surveillance via satellite ADS-B", Cairo, Egypt, Fév. 2015. En ligne : http://www.icao.int/RO_MID/Documents/2014/AFSWSandFSMP32/FSMP-WGF32-7-SpectrumWorkshop_GlobalATCSurveillanceviasatelliteABS-B.pdf

U.S. Government Publishing Office, "General operating and flight rules", Mars 2017. En ligne : https://www.ecfr.gov/cgi-bin/text-idx?node=14:2.0.1.3.10#se14.2.91_1225

M. J. Viggiano, E. M. Valovage, K. B. Samuelson, et D. L. Hall, "Secure ADS-B authentication system and method", U.S. Brevet US7 730 307, Jan., 2010. En ligne : http://www.google.com/patents/US7730307

J. A. Volpe, "Vulnerability assessment of transportation infratructure relying on the gloabl positionning system", U.S. Department of Transportation, Rapp. tech., Août 2001. En ligne : https://www.navcen.uscg.gov/pdf/vulnerability_assess_2001.pdf

W3C, "SPARQL Query Language for RDF", Jan. 2008. En ligne : https://www.w3.org/TR/rdf-sparql-query/

W. Wang, G. Chen, R. Wu, D. Lu, et L. Wang, “A low-complexity spoofing detection and suppression approach for ADS-B”, dans 2015 Integrated Communication, Navigation and Surveillance Conference (ICNS). IEEE, 2015, pp. K2–1. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7121236

K. D. Wesson, T. E. Humphreys, et B. L. Evans, “Can cryptography secure next generation air traffic surveillance ?” IEEE Security and Privacy Magazine, 2014. En ligne : https://radionavlab.ae.utexas.edu/images/stories/files/papers/adsb_for_submission.pdf

C. K. Wong et S. S. Lam, “Digital signatures for flows and multicasts”, dans Proceedings Sixth International Conference on Network Protocols (Cat. No.98TB100256), Oct. 1998, pp. 198–209. DOI : 10.1109/ICNP.1998.723740

G. Wright, “NAV CANADA implements ADS-B”, dans 2009 Integrated Communications, Navigation and Surveillance Conference. IEEE, 2009, pp. 1–9. En ligne : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5172868

H. Yang, R. Huang, X. Wang, J. Deng, et R. Chen, “EBAA : An efficient broadcast authentication scheme for ADS-B communication based on IBS-MR”, Chinese Journal of Aeronautics, vol. 27, no. 3, pp. 688–696, Juin 2014. DOI : 10.1016/j.cja.2014.04.028. En ligne : http://linkinghub.elsevier.com/retrieve/pii/S1000936114000983

ANNEXE A Format d'un message ADS-B de position en vol tel que défini par l'OACI (2008)

Table A-2-5. BDS code 0,5 — Extended squitter airborne position

MB FIELD

1	MSB	
2		FORMAT TYPE CODE
3		(specified in §A.2.3.1)
4		
5	LSB	
6	MSB	SURVEILLANCE STATUS
7	LSB	(specified in §A.2.3.2.6)
8	SINGLE ANTENNA FLAG (SAF) (specified in §A.2.3.2.5)	
9	MSB	
10		
11		
12		ALTITUDE
13		(specified by the FORMAT TYPE CODE)
14		
15		
16	This is (1) the altitude code (AC) as specified in	
17	§3.1.2.6.5.4 of Annex 10, Volume IV, but with the	
18	M-bit removed, or (2) the GNSS height (HAE)	
19		
20	LSB	
21	TIME (T) (specified in §A.2.3.2.2)	
22	CPR FORMAT (F) (specified in §A.2.3.2.1)	
23	MSB	
24		
25		
26		
27		
28		
29		
30		ENCODED LATITUDE
31		(CPR airborne format specified in §A.2.6)
32		
33		
34		
35		
36		
37		
38		
39	LSB	
40	MSB	
41		
42		
43		
44		
45		
46		
47		ENCODED LONGITUDE
48		(CPR airborne format specified in §A.2.6)
49		
50		
51		
52		
53		
54		
55		
56	LSB	

PURPOSE: To provide accurate airborne position information.

Surveillance status shall be coded as follows:

0 = No condition
1 = Permanent alert (emergency condition)
2 = Temporary alert (change in Mode A identity code other than emergency condition)
3 = SPI condition

Codes 1 and 2 shall take precedence over code 3.

When horizontal position information is unavailable, but altitude information is available, the airborne position message shall be transmitted with a format TYPE Code of ZERO (0) in bits 1 — 5 and the barometric pressure altitude in bits 9 to 20. If neither horizontal position nor barometric altitude information is available, then all 56 bits of transponder register 05_{16} shall be zeroed. The ZERO format TYPE Code field shall indicate that latitude and longitude information is unavailable, while the ZERO altitude field shall indicate that altitude information is unavailable.

ANNEXE B Format d'un message ADS-B d'identification tel que défini par l'OACI (2008)

Table A-2-8. BDS code 0,8 — Extended squitter aircraft identification and category

MB FIELD

1	MSB	
2		FORMAT TYPE CODE
3		(specified in §A.2.3.1)
4		
5	LSB	
6	MSB	
7		AIRCRAFT CATEGORY
8	LSB	
9	MSB	
10		
11		CHARACTER 1
12		
13		
14	LSB	
15	MSB	
16		
17		CHARACTER 2
18		
19		
20	LSB	
21	MSB	
22		
23		CHARACTER 3
24		
25		
26	LSB	
27	MSB	
28		
29		CHARACTER 4
30		
31		
32	LSB	
33	MSB	
34		
35		CHARACTER 5
36		
37		
38	LSB	
39	MSB	
40		
41		CHARACTER 6
42		
43		
44	LSB	
45	MSB	
46		
47		CHARACTER 7
48		
49		
50	LSB	
51	MSB	
52		
53		CHARACTER 8
54		
55		
56	LSB	

PURPOSE: To provide aircraft identification and category.

Note.— Since there is no internationally agreed criteria for wake vortex categorization, code 4 (set "A") is interpreted as indicating a medium category aircraft exhibiting higher than typical wake vortex characteristics.

Format type shall be coded as follows:

1 = Identification aircraft, category set D
2 = Identification aircraft, category set C
3 = Identification aircraft, category set B
4 = Identification aircraft, category set A

Aircraft/vehicle category shall be coded as follows:

Set A:

0 = No aircraft category information
1 = Light (< 15 500 lbs or 7 031 kg)
2 = Medium 1 (>15 500 to 75 000 lbs, or 7 031 to 34 019 kg)
3 = Medium 2 (>75 000 to 300 000 lbs, or 34 019 to 136 078 kg)
4 = High vortex aircraft
5 = Heavy (> 300 000 lbs or 136 078 kg)
6 = High performance (> 5g acceleration) and high speed (> 400 kt)
7 = Rotorcraft

Set B:

0 = No aircraft category information
1 = Glider/sailplane
2 = Lighter-than-air
3 = Parachutist/skydiver
4 = Ultralight/hang-glider/paraglider
5 = Reserved
6 = Unmanned aerial vehicle
7 = Space/transatmospheric vehicle

Set C:

0 = No aircraft category information
1 = Surface vehicle — emergency vehicle
2 = Surface vehicle — service vehicle
3 = Fixed ground or tethered obstruction
4 – 7 = Reserved

Set D: Reserved

Aircraft identification coding (characters 1 – 8) shall be:

As specified in Table A-2-32.

ANNEXE C Format d'un message ADS-B de vitesse en vol tel que défini par l'OACI (2008)

Table A-2-9a. BDS code 0,9 — Extended squitter airborne velocity (Subtypes 1 and 2: Velocity over ground)

MB FIELD

1	MSB		1	
2			0	
3	FORMAT TYPE CODE = 19		0	
4			1	
5	LSB		1	
6	SUBTYPE 1	0	SUBTYPE 2	0
7		0		1
8		1		0
9	INTENT CHANGE FLAG (specified in §A.2.3.5.3)			
10	IFR CAPABILITY FLAG			
11	MSB NAVIGATION UNCERTAINTY			
12	CATEGORY FOR VELOCITY			
13	LSB (NUC_R)			
14	DIRECTION BIT for E-W Velocity: 0 = East, 1 = West			
15	EAST — WEST VELOCITY			
16	NORMAL: LSB = 1 knot		SUPERSONIC: LSB = 4 knots	
17	All zeros = no velocity information		All zeros = no velocity information	
18	Value	Velocity	Value	Velocity
19	1	0 kt	1	0 kt
20	2	1 kt	2	4 kt
21	3	2 kt	3	8 kt
22	…	…	…	…
23	1 022	1 021 kt	1 022	4 084 kt
24	1 023	>1 021.5 kt	1 033	>4 086 kt
25	DIRECTION BIT for N-S Velocity: 0 = North, 1 = South			
26	NORTH — SOUTH VELOCITY			
27	NORMAL: LSB = 1 knot		SUPERSONIC: LSB = 4 knots	
28	All zeros = no velocity information		All zeros = no velocity information	
29	Value	Velocity	Value	Velocity
30	1	0 kt	1	0 kt
31	2	1 kt	2	4 kt
32	3	2 kt	3	8 kt
33	…	…	…	…
34	1 022	1 021 kt	1 022	4 084 kt
35	1 023	>1 021.5 kt	1 023	>4 086 kt
36	SOURCE BIT FOR VERTICAL RATE: 0 = GNSS, 1 = Baro			
37	SIGN BIT FOR VERTICAL RATE: 0 = Up, 1 = Down			
38	VERTICAL RATE			
39	All zeros = no vertical rate information; LSB = 64 ft/min			
40	Value		Vertical Rate	
41	1		0 ft/min	
42	2		64 ft/min	
43	…		…	
44	510		32 576 ft/min	
45	511		>32 608 ft/min	
46				
47	RESERVED FOR TURN INDICATOR			
48				
49	GNSS ALT. SIGN BIT: 0 = Above baro alt., 1 = Below baro alt.			
50	GNSS ALT. DIFFERENCE FROM BARO. ALT.			
51	All zeros = no information; LSB = 25 ft			
52	Value		Difference	
53	1		0 ft	
54	2		25 ft	
55	126		3 125 ft	
56	127		3 137.5 ft	

PURPOSE: To provide additional state information for both normal and supersonic flight.

Subtype shall be coded as follows:

Code	Velocity	Type
0	Reserved	
1	GroundSpeed	Normal
2		Supersonic
3	Airspeed,Heading	Normal
4		Supersonic
5	Reserved	
6	Reserved	
7	Reserved	

IFR capability shall be coded as follows:

0 = Transmitting aircraft has no capability for ADS-B-based conflict detection or higher level (class A1 or above) applications.

1 = Transmitting aircraft has capability for ADS-B-based conflict detection and higher level (class A1 or above) applications.

NUC_R shall be coded as follows:

NUC_R	Horizontal Velocity Error (95%)	Vertical Velocity Error (95%)
0	Unknown	Unknown
1	< 10 m/s	< 15.2 m/s (50 fps)
2	< 3 m/s	< 4.6 m/s (15 fps)
3	< 1 m/s	< 1.5 m/s (5 fps)
4	< 0.3 m/s	< 0.46 m/s (1.5 fps)

ANNEXE D Format d'un message ADS-B des prochains *waypoints* tel que défini par l'OACI (2008)

Table A-2-84 to A-2-86. BDS codes 5,4 to 5,6 — Waypoints 1, 2 and 3

MB FIELD

1	STATUS (see 1)
2	MSB
3	
4	CHARACTER 1
5	
6	
7	LSB
8	MSB
9	
10	CHARACTER 2
11	
12	
13	LSB
14	MSB
15	
16	CHARACTER 3
17	
18	
19	LSB
20	MSB
21	
22	CHARACTER 4
23	
24	
25	LSB
26	MSB
27	
28	CHARACTER 5
29	
30	
31	LSB
32	MSB = 30 minutes
33	
34	ESTIMATED TIME OF ARRIVAL
35	(NORMAL FLIGHT)
36	
37	Range = [0, 60] minutes
38	
39	
40	LSB = 60/512 minutes
41	MSB = 320 FL
42	
43	ESTIMATED FLIGHT LEVEL
44	(NORMAL FLIGHT)
45	Range = [0, 630] FL
46	LSB = 10 FL
47	MSB = 30 minutes
48	
49	TIME TO GO
50	(DIRECT ROUTE)
51	
52	Range = [0, 60] minutes
53	
54	
55	LSB = 60/512 minutes
56	RESERVED

PURPOSE: To provide information on the next three waypoints, register 54_{16} contains information on the next waypoint, register 55_{16} contains information on the next waypoint plus one, and register 56_{16} contains information on the next waypoint plus two.

1) The single status bit shall be set to ZERO (0) if any of the parameters are invalid.

2) The actual time or flight level shall be calculated from the trajectory scheduled in the FMS.

Note.— Mode detail on the next waypoint is given in register 41_{16} to 43_{16}.

3) When the waypoint identity has only three characters, two leading ZERO characters shall be added (e.g. CDN becomes 00CDN).

4) Estimated time is in minutes and all ones shall be used to indicate that the waypoint referred to is one hour or more away.

ANNEXE E Formulaire d'immatriculation d'un avion auprès de la FAA aux États-Unis

U.S. Department of Transportation
Federal Aviation Administration

UNITED STATES OF AMERICA – DEPARTMENT OF TRANSPORTATION
Federal Aviation Administration – Mike Monroney Aeronautical Center

OMB Control No. 2120-0042
Collection Expires 4/30/2017

AIRCRAFT REGISTRATION APPLICATION

UNITED STATES REGISTRATION NUMBER N	TYPE OF REGISTRATION (Check one box) ☐ 1. Individual ☐ 2. Partnership ☐ 3. Corporation (Includes LLC's) ☐ 4. Co-Owner ☐ 5. Government ☐ 8. Non-Citizen Corporation ☐ 9. Non-Citizen Corporation Co-Owner
AIRCRAFT MANUFACTURER AND MODEL	
AIRCRAFT SERIAL NUMBER	

NAME(S) OF APPLICANT(S) [Person(s) shown on evidence of ownership. If individual, give last name, first name and middle initial.]

TELEPHONE NUMBER: ( )

MAILING ADDRESS (Permanent mailing address for first applicant listed above.)

NUMBER AND STREET: _____

RURAL ROUTE: _____ P.O. BOX _____

CITY: _____ STATE: _____ ZIP: _____

PHYSICAL ADDRESS/LOCATION IF PO BOX OR RURAL ROUTE BOX USED FOR MA L NG ADDRESS

NUMBER AND STREET: _____

DESCRIPTION OF LOCATION: _____

CITY: _____ STATE: _____ ZIP: _____

☐ ***CHECK HERE IF YOU ARE ONLY REPORTING A CHANGE OF ADDRESS***

ATTENTION! Read the following statement before signing this application.
This portion must be completed.
A false or dishonest answer to any question in this application
may be grounds for punishment by fine and/or imprisonment.
(U.S. Code, Title 18, Section 1001)

CERTIFICATION

I/WE CERTIFY:

(1) That the above aircraft is owned by the undersigned applicant who either is a citizen (including corporations) of the United States **OR** meets the aircraft registration citizenship requirements of 14 CFR Part 47 as: (CHECK AND COMPLETE a, b, or c)

☐ a. A resident alien with alien registration (Form 1-551) No. _____

☐ b. A non-citizen corporation organized and doing business under the laws of (state) _____
and said aircraft is based and primarily used in the United States. Records of flight hours are available for Inspection at _____

☐ c. A corporation using a voting trust to qualify Enter name of trustee _____

(2) That the aircraft is not registered under the laws of any foreign country; and

(3) That legal evidence of ownership is attached or has been filed with the Federal Aviation Administration.

NOTE: If executed for co-ownership, all applicants must sign. Use next page if necessary.

1	SIGNATURE:	DATE:
	TYPED/PRINTED NAME:	TITLE:
2	SIGNATURE:	DATE:
	TYPED/PRINTED NAME:	TITLE:
3	SIGNATURE:	DATE:
	TYPED/PRINTED NAME:	TITLE:

NOTE: When carried in the aircraft with an appropriate current airworthiness certificate or a special flight permit, a copy of this completed application provides authority to operate the aircraft in the United States for up to 90 days.

ANNEXE F Exemple de certificat d'immatriculation de la FAA

REGISTRATION NOT TRANSFERABLE

UNITED STATES OF AMERICA
DEPARTMENT OF TRANSPORTATION - FEDERAL AVIATION ADMINISTRATION
CERTIFICATE OF AIRCRAFT REGISTRATION

This certificate must be in aircraft when operated.

NATIONALITY AND REGISTRATION MARKS: N 12345

AIRCRAFT SERIAL NO.: 6969

MANUFACTURER AND MANUFACTURER'S DESIGNATION OF AIRCRAFT
CESSNA C-150L
ICAO Aircraft Address Code:

ISSUED TO
ROBERT E. BARO
300 MOERKLE ST
ANYTOWN, OHIO
12345

This certificate is issued for registration purposes only and is not a certificate of title. The Federal Aviation Administration does not determine rights of ownership as between private persons.

It is certified that the above described aircraft has been entered on the register of the Federal Aviation Administration, United States of America, in accordance with the Convention on International Civil Aviation dated December 7, 1944, and with the Federal Aviation Act of 1958, and regulations issued thereunder.

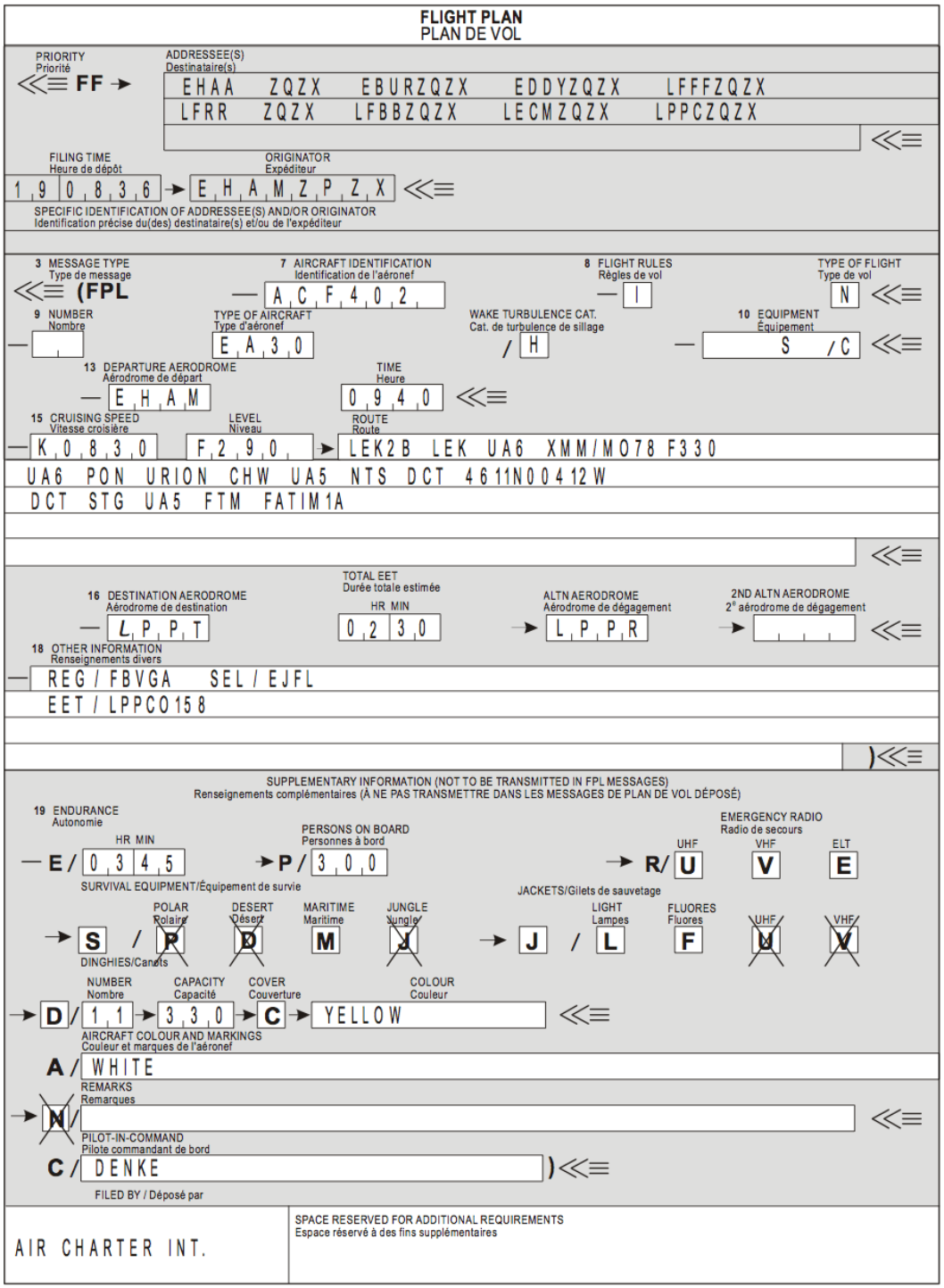
U.S. Department of Transportation
Federal Aviation Administration

DATE OF ISSUE
February 15, 1996

David Hinson ADMINISTRATOR

AC Form 8050-3(11/93) Supersedes previous editions

ANNEXE G Exemple de plan de vol sous le format standard défini par l'OACI



FLIGHT PLAN
PLAN DE VOL

PRIORITY / Priorité: FF
ADDRESSEE(S) / Destinataire(s): EHAA ZQZX EBURZQZX EDDYZQZX LFFFZQZX
LFRR ZQZX LFBBZQZX LECMZQZX LPPCZQZX
FILING TIME / Heure de dépôt: 190836
ORIGINATOR / Expéditeur: EHAMZPZX
SPECIFIC IDENTIFICATION OF ADDRESSEE(S) AND/OR ORIGINATOR
Identification précise du(des) destinataire(s) et/ou de l'expéditeur

3 MESSAGE TYPE / Type de message: (FPL
7 AIRCRAFT IDENTIFICATION / Identification de l'aéronef: ACF402
8 FLIGHT RULES / Règles de vol: I
TYPE OF FLIGHT / Type de vol: N
9 NUMBER / Nombre:
TYPE OF AIRCRAFT / Type d'aéronef: EA30
WAKE TURBULENCE CAT. / Cat. de turbulence de sillage: H
10 EQUIPMENT / Équipement: S /C
13 DEPARTURE AERODROME / Aérodrome de départ: EHAM
TIME / Heure: 0940
15 CRUISING SPEED / Vitesse croisière: K0830
LEVEL / Niveau: F290
ROUTE / Route: LEK2B LEK UA6 XMM/M078 F330
UA6 PON URION CHW UA5 NTS DCT 4611N00412W
DCT STG UA5 FTM FATIM1A

16 DESTINATION AERODROME / Aérodrome de destination: LPPT
TOTAL EET / Durée totale estimée HR MIN: 0230
ALTN AERODROME / Aérodrome de dégagement: LPPR
2ND ALTN AERODROME / 2e aérodrome de dégagement:
18 OTHER INFORMATION / Renseignements divers: REG/FBVGA SEL/EJFL
EET/LPPC0158

SUPPLEMENTARY INFORMATION (NOT TO BE TRANSMITTED IN FPL MESSAGES)
Renseignements complémentaires (À NE PAS TRANSMETTRE DANS LES MESSAGES DE PLAN DE VOL DÉPOSÉ)

19 ENDURANCE / Autonomie HR MIN: E/0345
PERSONS ON BOARD / Personnes à bord: P/300
EMERGENCY RADIO / Radio de secours: R/U (UHF) V (VHF) E (ELT)
SURVIVAL EQUIPMENT/Équipement de survie: S/ (POLAR/Polaire) (DESERT/Désert) (MARITIME/Maritime) M (JUNGLE/Jungle)
JACKETS/Gilets de sauvetage: J/L (LIGHT/Lampes) F (FLUORES/Fluores) (UHF) (VHF)
DINGHIES/Canots: D/ NUMBER/Nombre 11 CAPACITY/Capacité 330 COVER/Couverture C COLOUR/Couleur YELLOW
AIRCRAFT COLOUR AND MARKINGS / Couleur et marques de l'aéronef: A/WHITE
REMARKS / Remarques: N/
PILOT-IN-COMMAND / Pilote commandant de bord: C/DENKE
FILED BY / Déposé par: AIR CHARTER INT.
SPACE RESERVED FOR ADDITIONAL REQUIREMENTS / Espace réservé à des fins supplémentaires