A performance-aware Public Key Infrastructure for next generation connected aircrafts

Mohamed-Slim Ben Mahmoud, Nicolas Larrieu, Alain Pirovano

A PERFORMANCE-AWARE PUBLIC KEY INFRASTRUCTURE FOR NEXT GENERATION CONNECTED AIRCRAFTS

Mohamed Slim Ben Mahmoud, Nicolas Larrieu, Alain Pirovano
French Civil Aviation University (ENAC), LEOPART Laboratory, Toulouse, France

Abstract

This paper aims to illustrate the feasibility of a scalable Public Key Infrastructure (PKI) adapted for upcoming network-enabled aircrafts with a particular emphasis on the revocation and verification procedures: many techniques are discussed and their benefits in term of resulting overheads are underlined through a performance assessment study. The proposed PKI is also used to secure a negotiation protocol for the supported and common security mechanisms between two end entities. The PKI presented in this paper is a sub-task of an overall security architecture for the *FAST* (Fiber-like Aircraft Satellite Telecommunications) project, co-funded by the Aerospace Valley pole and the French government (Direction Générale de la Compétitivité, de l'Industrie et des Services – DGCIS, Fonds Unique Interministériel – FUI). The purpose behind the project is to demonstrate the feasibility of high-capacity aircraft-earth communications through a low-cost satellite antenna technology. The project federates both industrial (EADS Astrium, Axess Europe, Vodea and Medes) and academic/institutional (ISAE, ENAC, LAAS-CNRS, Telecom Bretagne) partners.

Introduction and Problem Statement

Characteristics of the Future Aeronautical Communication Systems

Over the last decade, safety and security have been considered as the highest priority concerns in the air transport industry. Although physical security remains the major issue in people's thoughts, researchers and experts concord in their concern to focus on digital information security for the future network-enabled aircrafts. This is due, partially, to the increasingly heterogeneous nature of air-ground communications (Air Traffic Services – ATS, Operational Control Services – AOC, and Aeronautical Passenger Communication Services – APC) and the expected shift from voice to data communications in future Air Traffic Management (ATM): the worldwide airspace will become more and more congested, as the traffic is forecast to increase steadily the next ten years. Consequently, European and American international programs such as SESAR [1] and NextGen [2] have been created to modernize the ATM and integrate innovative approaches to the aviation world.

Moreover, airlines aim to offer a better flight experience to passengers by deploying a variety of additional services, mainly through broadband Internet access and In-Flight Entertainment (IFE), while reducing design and maintenance costs of such disposals. Other services can be imagined such as duty free credit card purchasing or cellular phone usage. Consequently, the use of Commercially available Off-The-Shelf (COTS) components becomes necessary to maintain high efficiency and interoperability at reduced overall cost. Such evolutions in the civil aviation industry may engender many potential security threats which have to be carefully addressed.

PKI Considerations in Future ATM Systems

For this purpose, a PKI can be an effective solution to cope with these emerging security issues. PKI is usually defined as a set of practices, technologies, and policies involved in several processes such as deployment, management, storage, and revocation of public key certificates when asymmetric cryptography is used. The aim is to create a "chain of trust" for securing digital data and authenticating end entities. In ground-based networks, PKI's are often deployed whenever a large group of users must communicate securely without necessarily knowing or trusting each other directly

[1] Single European Sky ATM Research (SESAR) is the Single European Sky (SES) technological and operational program initiative to meet future capacity and air safety needs.
[2] NextGen is the American program for ongoing evolution of the American National Airspace System (NAS) from a ground-based system of ATC to a satellite-based system of ATM.

978-1-4244-6618-4/10/$26.00 ©2010 IEEE

(e.g. securing emails, remote access, or web applications). The PKI concept has been modified in many ways to take into consideration the management of public keys, certificates, or digital identities in different networks such as wireless or mobile (e.g. 3G, MANET) networks.

In the aeronautical context, some works have relied on PKI to secure communication protocols [1] or to address electronic distribution of airplane software [2], for instance. Recommendations and best practices are also being defined in the Air Transport Association (ATA) specification 42 "Aviation Industry Standards for Digital Information Security" document [3], proposed by the Digital Security Working Group (DSWG). The ATA DSWG group develops industry specifications to facilitate the implementations of information security practices and technologies to the civil aviation community. This document deals with digital identity management and specifies standard digital certificate profiles for the air transport industry. PKIs are also intended to be used in the future commercial connected aircrafts such as AIRBUS A350 and BOEING B787, where many digital applications are deployed either for cabin facilities, or AOC specific applications such as Electronic Flight Bag[3] (EFB) application.

However, with the increasing number of aircraft in the worldwide airspace, some scaling issues, not yet addressed, arise: long term forecast studies predict an average air traffic growth up to 3,5% per year between 2007 and 2030 [4]. Moreover, a single airplane is expected to carry out miscellaneous embedded end entities, ranging from avionic systems to on-board users (e.g. a passenger accessing to various Internet services). The 53th edition of the World Air Transport Statistics (WATS) document of the International Air Transport Association (IATA) [5] reported a worldwide passenger growth of +22.1% between 1999 and 2008: as the number of aircrafts/passengers/systems using security grows, it is apparent that the amount of key pairs and digital certificates handled by the PKI increases. Also, the management of the PKI credentials gets more complicated because of the typical constricted network capacity of air-ground technologies: both signaling and data messages induced by the PKI have to be performed at lower cost. Air-ground link will probably no longer be a problem in future since SATCOM technologies will offer high capacities for effective PKI processing, but retrieving large certificate revocation lists (CRLs) for instance, can be an issue if aircrafts do not use caching mechanisms onboard.

The certificate format is another aspect which needs to be investigated in details: certificate parameters have to be tailored to applications in which they are used (APC, AOC, and ATS) and to the certificate owner (aircraft, passenger, avionic system, etc). Also, aircraft networks are mobile communication systems, and then some mobility considerations are important when a PKI is used: since the aircraft should get seamless service before landing, mutual authentication with an entity of another airline, airport or domain should be possible. Because different aviation organizations may have different security policies in their own PKIs, complex inter-working and roaming schemes between the aircrafts, end entities, or airlines are required. In such a system, deploying a "classical" PKI model becomes a difficult task, then, a great challenge lies in finding a well-suited PKI for the next-generation connected aircrafts.

This paper aims to illustrate the feasibility of a novel PKI adapted for upcoming network-enabled aircrafts. This is a performance-aware model using a combination of hierarchical Certificate Authorities (CA) in order to minimize the air-ground exchanges caused by any PKI-related operational process (checking and revoking certificates, for instance).The PKI model we propose in this paper works across three levels: the first level is relevant to ground-CAs interactions. The second level is related to the communications between airline-CAs and subordinate CAs on each aircraft. The last level deals with the onboard users and the subordinate CAs. Different phases of the certification process and key management are also described. Online Certificate Status Protocol (OCSP) [6] and CRLs servers are discussed to emphasize their benefits in terms of resulting network and computation overheads. The PKI model is finally applied on an ad-hoc protocol we proposed in the FAST project for the negotiation of the commonly Supported Security Protocols (SSP) between two end entities.

[3] EFB is an electronic display system used to perform AOC flight management tasks and intended to replace paper-based document used by the crew.

Nomenclature

Table 1 contains the notations used in the following sections:

Table 1. Notations

Notation	Description
K_i^+	The public key of an entity *i*
K_i^-	The private key of an entity *i*
N_C	Total number of certificates
N_f	Flight number at time *t*
$Size_C$	Average size of a certificate
t_C	Certificate validity period (in days)
t_S	SSP validity period (in days)
h_S	Digest using a hash function
$Nonce_i$	i^{th} randomly generated number
l_{sig}	Digital signature length
l_{sn}	Certificate serial number length
C_{sig}	Signature generation time
C_v	Signature verification time
M	Exchanged data
$\{i, K_i^+\}_{K_{CA}^-}$	Certificate of *i* issued by *CA*
R_C	% of revoked certificates
N_R	Certificate revocation check status messages per day
N_U	Revocation information update messages per day
$N_{C,CA}$	Certificate average number handled by one *CA*
C_U^{Net}	Network cost to update a certificate between *CA* and *CMSE*[4]
$C_{U,CA}^{CPU}$	Computation cost at *CA* to update a certificate
$C_{U,CMSE}^{CPU}$	Computation cost at *CMSE* to update a certificate
C_R^{Net}	Network cost to check a certificate between *CMSE* and a *verifier*
$C_{R,CA}^{CPU}$	Computation cost at *CA* to check a certificate
$C_{R,CMSE}^{CPU}$	Computation cost at *CMSE* to check a certificate
$C_{R,V}^{CPU}$	Computation cost at *verifier* to check a certificate

[4] CMSE: *Certificate Management Subordinate Entity, see section "Hierarchical PKI Model for Next Generation Connected Aircrafts" for details.*

Introduction to Basic PKI Concepts

In this section, we present a non exhaustive overview of the basic PKI concepts commonly used. More details about PKIs can be found in [7].

Security Services

A PKI is intended to offer the following security features:

- *Confidentiality of communications*: only allowed persons will be able to read encrypted messages;
- *Non repudiation*: the sender cannot deny to a third party that he sent a given message;
- *Integrity of communications*: the recipient of a message is able to determine if the message content was not altered during its exchange;
- *Authentication of the sender*: the recipient is able to identify the sender of a message and to demonstrate to a third party, if required, that the sender was properly identified.

PKI Cryptographic Resources

When a PKI is deployed, fundamental cryptographic elements are used:

- *Public and private keys*: also known as asymmetric key pairs. Every end entity holds two keys; the public key is made publicly available to all the other entities of the system while the private key is kept secret. The keys are one-way functions, which means it is considerably difficult to decrypt a message if it has been encrypted with one of the two keys. Also, the keys are mathematically related: if a message *M* is encrypted using the public key K_i^+, only the private key K_i^- allows us to reveal the message:

  $$\{\{M\}_{K_i^+}\}_{K_i^-} = M$$

  The reciprocal function is also true: if *M* is encrypted with the private key K_i^-, the public key K_i^+ is used to find the message:

  $$\{\{M\}_{K_i^-}\}_{K_i^+} = M$$

RSA (Rivest, Shamir, Adleman) [8] is a well-known asymmetric algorithm based on public/private keys cryptography;

- *Digital Certificates*: this is a central element in the use of asymmetric key pair's technique. A certificate is a data structure used to bind a public key to an end entity in an authentic way. The certificate has to be signed by a trusted third party (cf. PKI entities below) and it ensures that the public key really belongs to the entity that is stated in the certificate. A certificate aggregates many information such as a unique certificate number, the issuer identifier, the owner identifier, the public key, the algorithm used to generate the signature or a validity period. Other information fields can be included, depending on the type and the purpose of the certificate. The ITU-T X.509 format is the most known and widely used certificate; in Internet applications [9];
- *Hash values*: (also known as checksums or digests), a hash value is a piece of data computed using a hash function. A hash function is a mathematic function which takes a variable size data and returns a fixed size value. When used in cryptography, a hash function has to be one-way (computationally hard to invert), collision free (computationally impossible to find the same hash for two different data inputs), and fixed length output (the function has to produce always the same size data length). SHA-1 (Secure Hash Algorithm) [10] is an example of a hash function which can be used to compute 160 bits length hashes. In PKI, hashes are used to produce digital signatures;
- *Digital signatures*: a digital signature is the output of a cryptographic process used to certify the signer identity and also the integrity of the data being signed. A digital signature is produced as follow: a checksum is computed then encrypted using the private key K_i^- of the signer. The resulting digital signature is added to the signer's certificate and attached to the signed data. In order to verify a digital signature, the first condition is the validity of the signer's digital certificate (i.e. not expired and not revoked). A relying party decrypts the signature using the public key K_i^+ of the signer (bound to the certificate) to get the signer's hash value. Then, the relying party computes himself the hash of the data and compares the two hashes; if they match then data integrity can be assumed.

PKI Components

A PKI is composed of the following entities:

- *Certification Authority* (CA): this is the core component of a PKI since it is the only entity that can issue public key certificates. Any digital certificate is signed by the issuing CA, which makes it the very foundation of a security architecture using a PKI. If CRLs have not been delegated to an autonomous CRL issuer, CAs can also be responsible of issuing the CRLs;
- *Registration Authority* (RA): this is an optional component that verifies the users' identity and requests the CA to issue an adequate digital certificate;
- *End Entities:* an end entity is a generic term used to denote a user, a device or a piece of software that need a digital certificate. In the aeronautical context, an end entity can be a passenger, an aircraft, an airline or an operator for instance;
- *Repository*: this is also an optional component since it denotes a storage device for certificates and CRLs so that they can be retrieved by end entities.

Certificate Life Cycle Management

The management of certificate life cycle is the primary function of a PKI; the main steps are the following:

- *Registration and public/private keys generation (RK)*: the first step is the end entity registration and identity establishment. The registration procedure depends on which component has to generate the public/private keys. If the CA generates the key pair then the private key

is securely passed to the registering end entity through an Out-Of-Band [5] (OOB) mechanism, if the end entity generates the key pair, then the public key is passed to the CA which checks the validity of the private key by means of proof mechanisms. The digital signature, which is generated using the private key and verified using the corresponding public key, can be such a mechanism;

- *Certificate generation and distribution (CGD)*: after the end entity registration and key pair generation, a certificate is issued and distributed respectively to the end entity and the certificate repository;
- *Certificate regeneration (CRG)*: when a certificate expires, the corresponding end entity informs the CA which has to renew the certificate;
- *Certificate revocation (CRV)*: when a private key has been compromised, the certificate is no longer valid and has to be revoked;
- *Certificate retrieval (CRT)*: end entities retrieve certificates from the repository or may exchange certificates between each other (when the Pretty Good Privacy[6] PGP is used, for instance [11]);
- *Certificate validation (CV)*: end entities may retrieve the CRLs from a repository or may connect to an OCSP server to validate a certificate when needed.

Figure 1 shows how all the PKI components interoperate which each others.

The performance analysis we made focused on two most important certificate life cycle management steps: generation/distribution and revocation certificate processes.

In order to highlight the advantages of our PKI model; we describe in the following section the most used certificate revocation schemes with more details.

[5] OOB can be offline or using a secure and trusted channel

[6] PGP is a protocol used to enhance the security of e-mail communications by providing cryptographic privacy and authentication mechanisms for exchanged data.

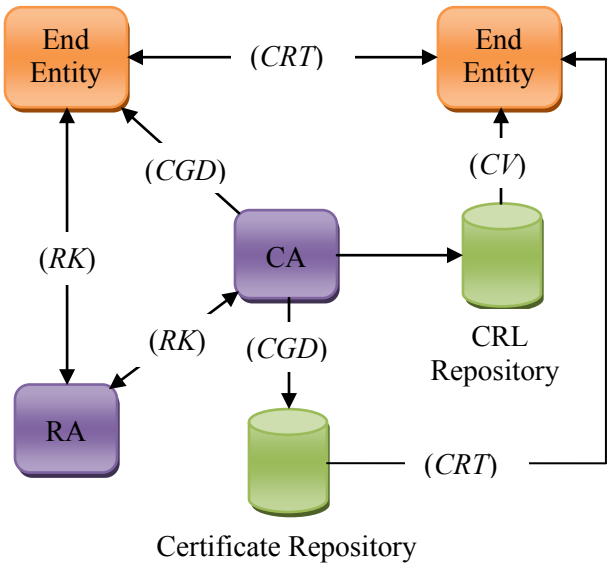

Figure 1. Basic PKI Environment

Certificate Revocation Schemes

Certificate validation is the process of verifying that a certificate is still valid: the validity period is checked and the process performs an integrity check based on the signature of the issuing CA and the revocation status to ensure that the certificate has not been revoked.

Certificate revocation is a different process since it is the action of declaring a certificate invalid before its expiration. For instance, the certificate revocation is required when the private key is compromised: the certificate becomes useless since the public key attached to it is mathematically related to the private key.

In a safety-related context such as data link communications, we think that the certificate revocation is an important process in the certificate cycle life management: any implemented PKI has to necessarily deploy a mechanism for revoking certificates and inform all involved entities about the certificate status. There are several approaches to revoke a certificate. The traditional technique is to publish a CRL containing all the revoked certificates ID's periodically.

The shortcoming of this approach is that the list size grows for large domains with many end entities downloading the list, and thus the network load becomes really heavy and unacceptable. Cache techniques can be used at the end entities, but it is

difficult to define the frequency of CRL updates and get a list as fresh as possible. Many modifications and extensions for improving CRL performances were proposed such as Delta CRL, Over-issued CRL or CRL distribution points [9].

The second standardized approach is to provide an online server and use some protocols to check in real-time the certificate revocation status. Compared to the CRLs, the main advantage is to request a targeted certificate status instead of a full revocation lists where only one entry matters for the verifier. OCSP is an example of an online revocation status checking protocol. The protocol has been designed to check the revocation status exclusively: an end entity requests the revocation information for one or more certificates using OCSP request to the OCSP server. The OCSP responder checks the revocation status information and issues an OCSP response containing the certificate ID and the certificate status to the end entity.

The problem with this approach is that the server response has to be signed (which means processing and network overheads for each response). Another issue is that the server is always connected, which makes it vulnerable to Denial of Service (DoS) attacks. As for CRLs, there are some proposals to add functionalities to OSCP and avoid this kind of issues such as OCSP-X [12]. Simple Certificate Verification Protocol (SCVP) [13] is another online protocol but little bit different from OSCP since it fully validates a certificate using all certificate validation criteria (expiration lifetime, issuer ID, etc). Since the classical CRLs and the online OCSP protocol are the two revocation mechanisms recommended in ATA Spec42 document [3], we perform a comparative analysis using only these two revocation schemes, but the study can be extended to other revocation mechanisms in further work.

PKI Activities in Civil Aviation

Many research works have been carried on PKIs to enhance the security of next generation connected aircrafts. For instance, [14] investigated an authentication protocol for Controller-Pilot Data Link Communications (CPDLC). As far as public keys and certificates are needed (the protocol is based on elliptic curve primitives), a PKI was used and the authors assumed that a CA exists to create and distribute the credentials between the aircraft applications and the ground-CPDLC applications. But, there were no cost or performance considerations when the PKI was presented. Moreover, the PKI described here is specific to one particular protocol.

[1] proposed a secure version of the Aircraft Communications Addressing and Reporting System (ACARS). ACARS system is worldwide used by commercial airlines for the air-ground operational communications and over oceanic regions when radar coverage is no longer available. The messages are transferred over Radio Frequency (RF) channels in readable forms: then, it is possible to determine aircraft position, cargo content or operational details of the flight using low cost eavesdropping techniques[7]. The AMS (ACARS Message Security) protocol is a security solution for ACARS and uses cryptographic algorithms that have been validated by the industry, such as PKI for the key and certificate management life cycle. Unfortunately, The ACARS is intended to be replaced progressively over the years with the ATN (Aeronautical Telecommunication Network) over IPS (Internet Protocol Suite) system.

Besides, the use of data networks creates some opportunities to corrupt safety-critical airplane software's: [2] presented a security framework for a specific aeronautical network application, namely the Electronic Distribution of Software (EDS). First, the authors introduced a new approach called Airplane Assets Distribution system (AADS) to model the information assets exchange between the entities. They identified safety and business threats, then suggested to use digital signatures and a PKI to secure the model, but they considered the PKI security solution too much complex (because of the certification mandatory procedure) and they proposed to investigate a light-weight alternative to PKI.

[15] addresses some of the emerging challenges for network-enabled airplanes that use public key cryptography-based applications. Two approaches have been presented, respectively an ad-hoc technique without trust chains between certificates, and a structured approach employing a PKI for EDS on commercial airplanes. The ad-hoc approach

[7] Acarsd is a free ACARS decoder for Linux and Windows OS which attempts to decode ACARS transmissions in real-time using soundcard devices.

consisted in pre-loading trusted certificates on airplane via an OOB mechanism: the main advantage of the solution is its simplicity and reduced cost, the big drawback is the fact that this solution does not consider the scaling issues we discussed before. The structured PKI solution seems much more appropriate and offers long-term benefits in terms of scalability. But it is considered more expensive than the ad-hoc solution, specifically because of the setting up and maintenance costs of the PKI.

The paper discussed also the certificate revocation main techniques: the authors suggested using CRLs for checking certificates at the airplane to avoid the necessity of direct connectivity to external networks, which is a condition imposed by the use of an OCSP online server. In our study, we evaluate these techniques according to the induced network and computational overheads and we give suggestions for design and implementation according to the results obtained at the end of our paper.

[16] depicts in general how a PKI supports an ATM environment with an emphasis on the ATN and the Federal Aviation Administration (FAA) facilities and devices (routers, end systems, LANs, etc). The authors suggested the use of cross certification to handle inter-domain certification. Cross certification is basically an extension of the third party trust term: when different CAs are deployed in separate domains, they are able to establish and maintain trustworthy certification processes and interact seamlessly with each other. In this way, users are able to trust any CA as much as their own CA and can communicate with users not necessarily attached to the same CA. However, the key distribution and certification processes were not described in this paper.

Hierarchical PKI Model for Next Generation Connected Aircrafts

Standard PKI Model

Several types of PKI have been defined for ground networks [17]. As a reference (and because it is widely deployed model), we have chosen a single CA model as a standard PKI model for the performance study.

Figure 2 shows the entities involved with a single root CA, these CAs are deployed by the airlines on the ground:

- *Verifier*: an end entity which aims to verify the validity of a certificate;
- *Owner*: an end entity which possesses the digital certificate to be verified;
- *Certificate Management Subordinate Entity*[8] *(CMSE)*: this is an entity through which the verifier is able to check the certificate status and its validity (e.g. an OCSP server). In most case, the CMSE is merged with the CA.

It is important to note that both verifier and certificate owner can be either onboard or ground-located. The computation and network overheads are also depicted in figure 2 (cf. table 1 for the descriptions of notations). The owner is denoted *O*, the verifier *V*, and the ground CA *GCA* in the equations.

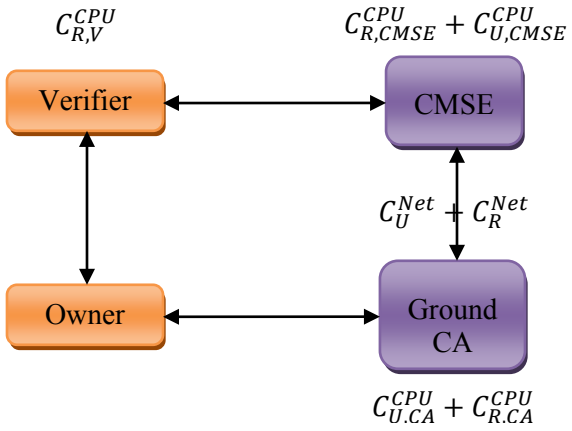

Figure 2. Standard PKI model

Hierarchical PKI Model

In this section, we propose a PKI model adapted to the future aeronautical air-ground communications. Figure 3 illustrates the model and the function of each entity:

[8] Depending on the used terminology, CMSE might have a different name.

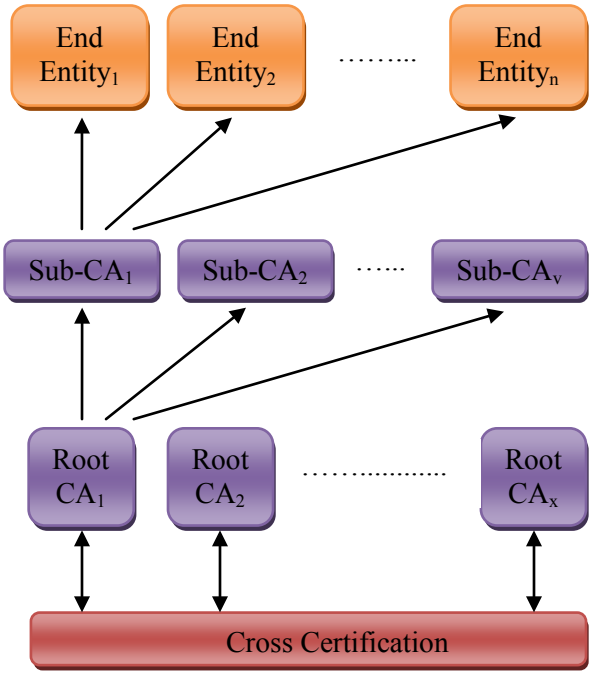

Figure 3. Hierarchical PKI Model for Future Aeronautical Communications

The PKI model we propose works across three levels:

- The first level is relevant to the inter-CA communications: a ground-located root-CA (*RCA* in the equations) is deployed for each airline and is responsible of all the end entities that belong to this airline. The end entity can be on the ground such as an ATN router (out-of-scope of this paper) or an aircraft (see the second level of the hierarchy). As long as every root-CA is independent of the others and has the authority on the aircrafts labeled within the airline domain, cross certification can be used between the root-CAs. Thus, the autonomy of local ground CAs and interaction between end entities belonging to different airlines can be always provided;
- The second level is relevant to the communications between the root CA of an airline and the aircrafts managed by this root-CA: delegated (or subordinate) CAs (denoted *SCA* in the equations) are deployed onboard each aircraft and used to handle the onboard certificate entities (see the last level of the hierarchy). Actually, using a device as a CA in mobile networks is of common use, especially for performance purposes (in MANETs for instance): we used this idea as a starting point to develop our scalable PKI model;
- The third and last level of the hierarchy concerns every end entity onboard the aircraft: the sub-CA is responsible of managing all the certificates of these entities. In the analysis performed below, only passengers are considered as end entities holding a certificate, but the study can be extended to avionic devices or AOC crew for instance.

Performance Analysis

In this section, we compare the two PKI models in three different study cases; depending on the verifier, the certificate owner and the CAs physical locations (ground to ground case is out-of-scope of the study since there are no messages exchanged on the air-ground link). The comparison study is done for two PKI steps: the certificate generation and revocation procedures. The main goal of the study is to evaluate network and computation overheads generated by the different PKI models according to the physical locations of PKI entities defined for each scenario.

Aircraft Source Data

Our study is passenger-based approach, which means we rely exclusively on the number of growing passengers to evaluate the benefits of the proposed model. For this purpose, it is adequate to use real data for the performance study: then we managed to use source traffic data issued from the DSNA-DTI (Direction des Services de la Navigation Aérienne-Direction de la Technique et de l'Innovation) databases. These are daily air traffic statistics for medium-range aircrafts in the French airspace and are structured by hour of flight, aircraft family label (e.g. B738), and ICAO (International Civil Aviation Organization) code.

In order to make these information more useful, we tried to estimate the maximum number of passengers that every aircraft can carry, and then we extrapolate the results by the total number of

aircrafts. We used The EUROCONTROL performance database [9] V2.0 and some additional information about aircraft seats [10] to deduce the maximum capacity of each aircraft according to its ICAO code, then we synthesize the data and extract the relevant information we need. Also, as suggested by a recent DGAC [11] (Direction Générale de l'Aviation Civile) report [18], we used an average aircraft filling (between 70% and 80%) instead of the maximum aircraft capacity. Also, as we used to deploy an airline-dedicated PKI (cross-certification between the airlines is out-of-scope of this paper), we concentrate our efforts on the largest's airline in the source data, namely the French *Air France* airline.

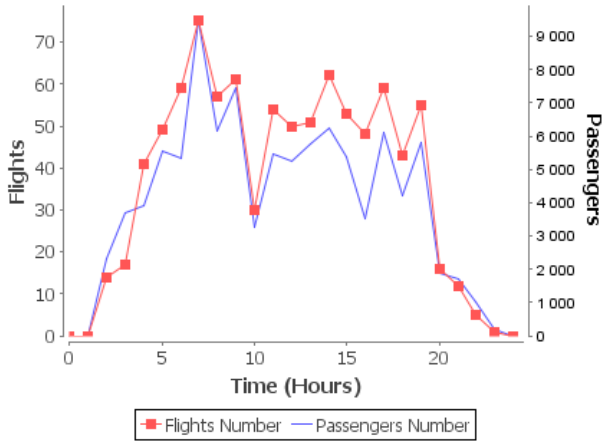

Figure 4. Daily Passenger and Aircraft Statistics (*Air France* Airline)

Figure 4 shows the global number of flights handled per hour (an average of 38 aircrafts) and the total passenger's number per hour (an average of 4200 passengers). These statistics will be used later to study the certificate management procedures and the network and computational costs.

Experimental Scenarios

Scenario 1: Ground-Verifier/on Board-Owner

This is a typical case where a passenger sends an email (signed) to a ground entity which wants to proceed for certificate verification. Figure 5 and figure 6 shows respectively the exchanged data in this scenario for the two PKI models. The dashed line is the air-ground separation.

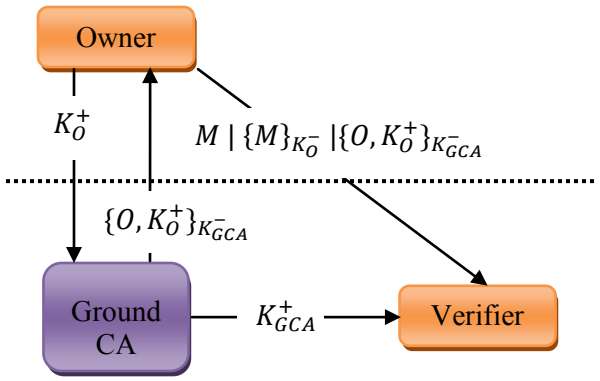

Figure 5. Scenario 1 – Standard Model

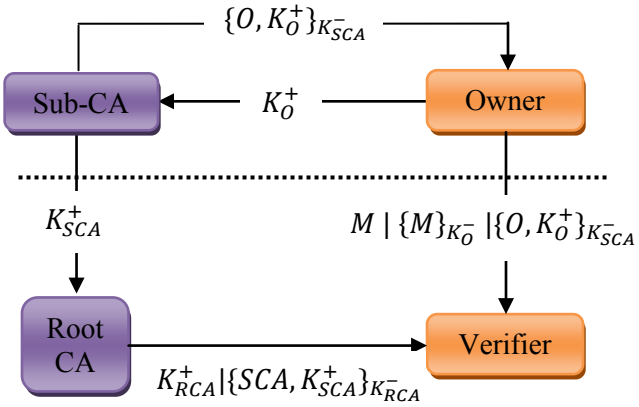

Figure 6. Scenario 1 – Hierarchical Model

Scenario 2: On Board-Verifier/Ground-Owner

In this scenario, the certificate owner (e.g. an email sender) is on the ground and the verifier is on board (see figure 7 and 8):

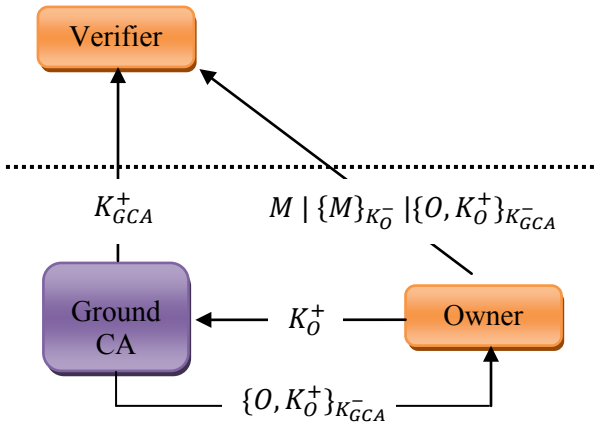

Figure 7. Scenario 2 – Standard Model

[9] www.elearning.ians.lu/aircraftperformance/
[10] www.seatguru.com
[11] The DGAC is the French civil aviation authority.

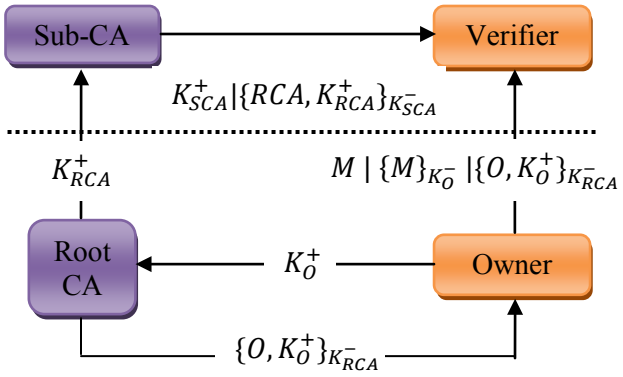

Figure 8. Scenario 2 – Hierarchical model

Scenario 3: Both Verifier and Owner Are on Board

In the last scenario, the verifier and the owner are both on board two different aircrafts as shown in figure 9 and figure 10. Intra-airline AOC information exchange can be a direct application of this specific scenario:

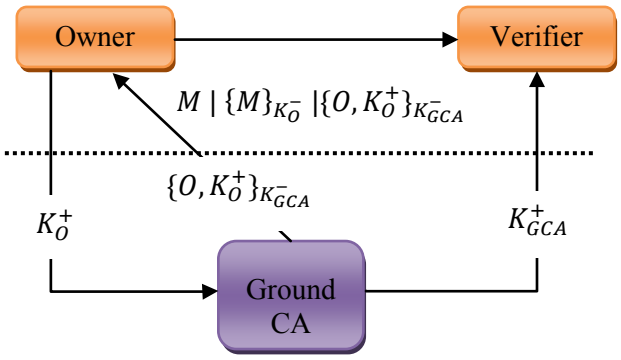

Figure 9. Scenario 3 – Standard Model

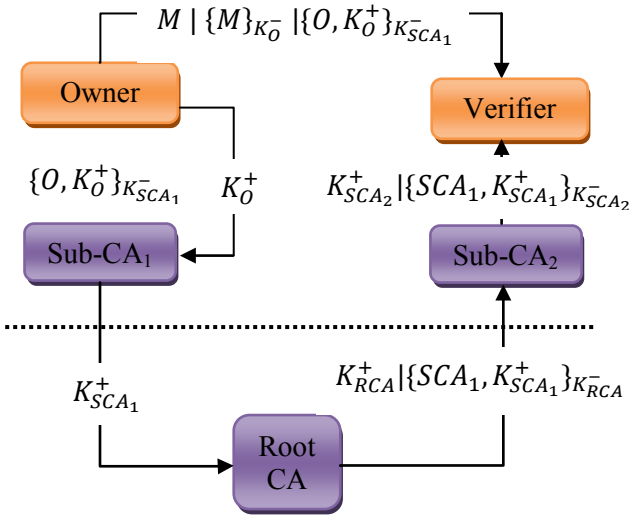

Figure 10. Scenario 3 – Hierarchical Model

Results

Certificate Generation and Distribution Process

In order to assess the network and the processing costs according to the two PKI models and the three different scenarios previously introduced, some assumptions have to be made:

- RSA is used for the key pairs and the digital signature with a signature key length $l_{sig} = 256\ Bytes$. For simplicity matter, we use l_{sig} notation to denote simultaneously the signature length and the public key length.
- The average certificate length is $Size_C = 1\ KBytes$ (based on the average X.509 certificate length);
- The exchanged data M is not considered since the study aims to measure only the additional overheads of PKI mechanisms.

Here are the two network cost equations respectively for the standard and the hierarchical PKI models (scenario 1):

$$N_C.\left(K^+_O + \{O, K^+_O\}_{K^-_{GCA}} + M \mid \{M\}_{K^-_O} \mid \{O, K^+_O\}_{K^-_{GCA}}\right) \cong 2.N_C.(l_{sig} + Size_C)$$

and

$$N_f.K^+_{SCA} + N_C.(M \mid \{M\}_{K^-_O} \mid \{O, K^+_O\}_{K^-_{SCA}}) \cong N_f.l_{sig} + N_C.(l_{sig} + Size_C)$$

The passenger is assumed to send one request for the certificate generation. We extrapolate the equations with the results we obtained from the aircraft and passenger statistics (cf. Aircraft Source Data Section):

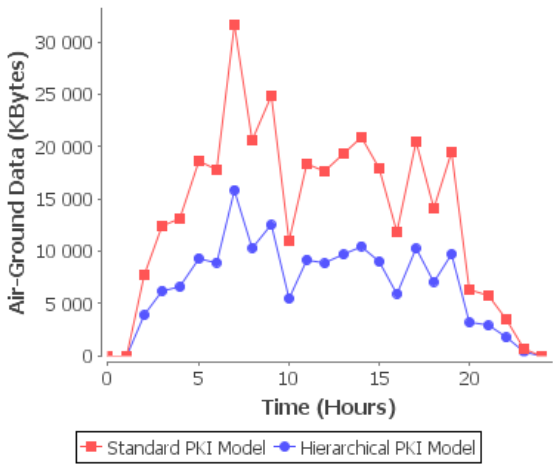

Figure 11. Scenario 1 – Network Costs

As shown in figure 11, it is clear that the hierarchical PKI model is less greedy than the standard model; the difference between the two model costs is about 55%. The hierarchical model is also better in the scenario 2 configuration, the network cost equations for the standard and hierarchical PKI models are:

$$N_C.\left(K^+_{GCA} + M\,|\{M\}_{K^-_O}\,|\{O, K^+_O\}_{K^-_{GCA}}\right) \cong N_C.\left(2.l_{sig} + Size_C\right)$$

and

$$N_f.K^+_{RCA} + N_C.\left(M\,|\{M\}_{K^-_O}\,|\{O, K^+_O\}_{K^-_{RCA}}\right) \cong N_f.l_{sig} + N_C.\left(l_{sig} + Size_C\right)$$

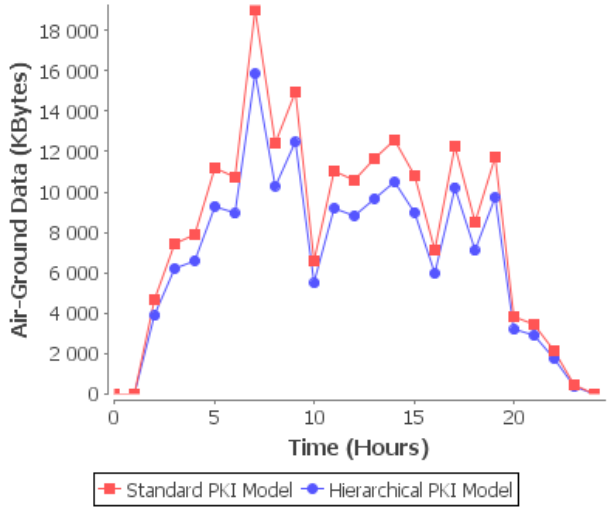

Figure 12. Scenario 2 - Network Costs

Figure 12 illustrates the network costs; the difference between the two PKI models is 20%. In the last scenario, the network cost equations are:

$$N_C.\left(N_f - 1\right).\left(K^+_O + \{O, K^+_O\}_{K^-_{GCA}} + K^+_{GCA} + M\,|\{M\}_{K^-_O}\,|\{O, K^+_O\}_{K^-_{GCA}}\right) \cong N_C.\left(N_f - 1\right).\left(3.l_{sig} + 2.Size_C\right)$$

and

$$\left(N_f - 1\right).\left(K^+_{SCA_1} + K^+_{RCA}\left|\{SCA_1, K^+_{SCA_1}\}_{K^-_{RCA}}\right.\right) + N_C.\left(M\,|\{M\}_{K^-_O}\,|\{O, K^+_O\}_{K^-_{SCA_1}}\right) \cong \left(N_f - 1\right).\left(2.l_{sig} + Size_C\right) + N_C\left(l_{sig} + Size_C\right)$$

The hierarchical model network cost remains always below the standard model network cost as we can see in figure 13. We used a logarithmic scale for this figure to see better the difference between the two models: the average difference for network costs is about 92 % per hour for all the passengers.

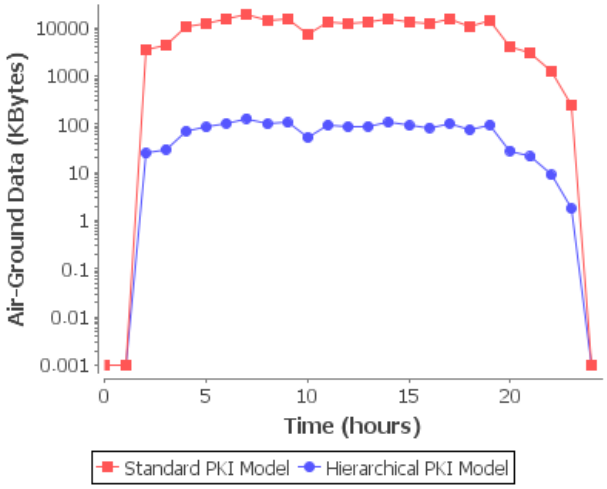

Figure 13. Scenario 3 - Network Costs

As we can see in both cost equations and figures, the hierarchical model is advantaged thanks to the number of total certificates that a root-CA has to manage; the deployment of the sub-CA minimizes the air-ground exchanges for the PKI credentials (public keys, signature and certificates). In the standard model, all these credentials are handled by a single ground-located CA, and then the air-ground amount of data is much larger.

Certificate Revocation Process

In this section, we analyze the same comparison study (using the same scenarios for both the standard and the hierarchical PKI models) regarding the revocation process using two techniques: CRLs and

OCSP protocol. Table 2 shows the value of each cost per revocation mechanisms:

Table 2. Network and Processing Costs for the Certificate Revocation Procedure

Cost	CRL	OCSP
C_U^{Net}	$N_U.(\frac{N_C.R_C.t_C.l_{sn}}{2}+\frac{N_C.l_{sig}}{N_{C,CA}})$	0
$C_{U,CA}^{CPU}$	$N_U.C_{sig}$	0
$C_{U,CMSE}^{CPU}$	$N_U.C_v$	0
C_R^{Net}	$N_R.(\frac{N_{C,CA}.R_C.t_C.l_{sn}}{2}+l_{sig})$	$N_R.l_{sig}$
$C_{R,CA}^{CPU}$	0	0
$C_{R,CMSE}^{CPU}$	0	$N_R.C_{sig}$
$C_{R,V}^{CPU}$	$N_R.C_v$	$N_R.C_v$

As for the certificate generation process, we make some assumptions on the parameters used in the certificate revocation performance study:

- A passenger holds only one certificate and then the total number of certificates N_C is equal to the total number of passengers (per hour);
- N_R (the certificate revocation check status messages per day) depends on the total number of certificates: $N_R = N_C.R_C$, where $R_C = 10\%$;
- $N_{C,CA}$ depends on the considered PKI model: in the standard model, $N_{C,CA} = N_C$ (equal to the total number of passengers per airline), in the hierarchical PKI model, $N_{C,CA} = 110$ (average passengers per sub-CA);
- Revocation information update frequency is one day: $N_U = 24\ (hours)$;
- RSA is always used for the key pairs and the digital signature: $l_{sig} = 256\ Bytes$;
- The certificate serial number length $l_{sn} = 20\ bits$;
- The signature and verification time's C_{sig} and C_v are respectively equal to *420 msec* and *0.113 msec*. These values are processed using a Pentium 8x Core i7 CPU at 2.67 Ghz, 4Go RAM and a Linux 2.6.26-2-64 kernel.

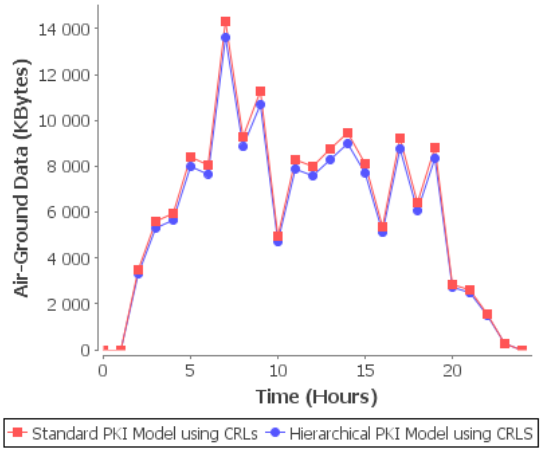

Figure 14. Requested Network Capacity between CA and CMSE for Updating Certificate Revocation Information

The CRLs are heavy and, then the update operation is expensive for the two PKI models: the difference is not significant. The OSCP approach is not represented in figure 14 because the server is usually co-located with the CA and then the requested network capacity is null. The computational cost of the CRL approach is really weak (up to *48 msec*), for OCSP this cost is null.

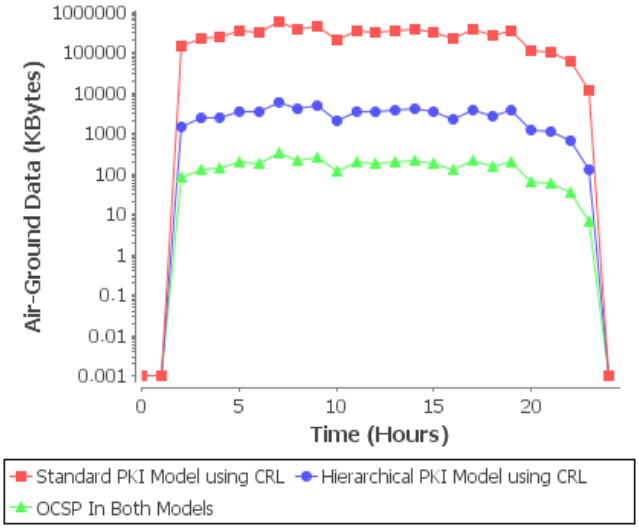

Figure 15. Requested Network Capacity between CA and Verifiers for Revocation Requests

The benefits of the hierarchical PKI model are much clearer when the comparison is done for the revocation request messages: the standard model is disadvantaged because of the total number of certificates handled by one ground CA. For the

hierarchical PKI model, OCSP is better than the classic CRL approach: OCSP computes only one signature per request whereas the CRL method is much more demanding in term of network capacity (cf. Figure 15). The computational costs are nearly the same except a difference for OCSP server (up to *9ms* versus *0 ms* for the CRL).

As expected, the hierarchical PKI has better performances than the standard PKI model. The CRL revocation method has many advantages such as its simplicity, an important amount of information, and a reduced risk. But, as shown in the experiments, the big size of the CRLs is a major issue since the requested network capacity for updating and checking the status of the certificates is extremely high. Also, for freshness purposes, every CRL contains the next update date of the revocation information: since all the verifiers are going to send CRL requests at the same time to retrieve the new CRL, the network might be overloaded at this time. These consequences cannot be accepted in the aeronautical context were the air-ground network resources cannot be wasted, thus, we recommend the use of OCSP as a revocation method instead of the CRL classic approach.

Securing a Negotiation Protocol of Supported Security Mechanisms

In a previous work, we introduced a negotiation protocol as a component of a whole security framework for aeronautical data link communications [19]. The aim of the proposal is to provide an adaptive security policy for APC, AOC, and ATS communications. A component called *SecMan* (Security Manager) is designed to pick up the best security mechanism, depending on real-time network and computational considerations. For the initiation of the adaptive algorithm, the onboard and ground servers have to negotiate the ciphers commonly supported before a secure connection can be established. Thus, we designed a negotiation protocol of the supported security mechanisms for air-ground communications. Initially, we proposed an unsecure version of the protocol, but quickly, we realized that the protocol was subject to many critical attacks such as replay and Man in The Middle (MITM) attacks. Then, we propose to use the PKI model in order to secure this negotiation protocol.

As an extension of the performance study discussed in this paper, we perform here the same comparison between the standard PKI model and the hierarchical PKI model. In this paper, we do not need to explain all the steps of the negotiation phase; the protocol is detailed in [19]. Instead, we focus only on the air-ground messages exchanged between the onboard security proxies (called SMP – Security Manager Proxy) and the ground server: if a passenger requests for a secure connection with a ground-located server, the SMP takes the lead and makes the negotiation with the server. In order to respect the terminology used above, the SMP is the *verifier* and the ground server (noted *S*) is the certificate *owner*. This case study is relative to the second scenario described before (an onboard *verifier* and a ground *owner*). For simplicity matter, the study is done only for the initiation phase of the negotiation protocol since the PKI credentials are mainly used in this step. Here are the numerical values used for the study:

- The Supported Security Protocols (SSP) set (added to its lifetime t_S) is stored in XML files and has a size equal to $400\ Bytes$;
- The hash h_S is generated using SHA-1 and has a $20\ Bytes$ length;
- The *Nonce* size is equal to $16\ Bytes$;
- RSA is used for the digital signature with a signature key length $l_{sig} = 256\ Bytes$;
- Certificate length is equal to $1\ KBytes$.

Figure 16 and 17 depicts the exchanged messages of the initial negotiation protocol phase using respectively the standard and hierarchical PKI models:

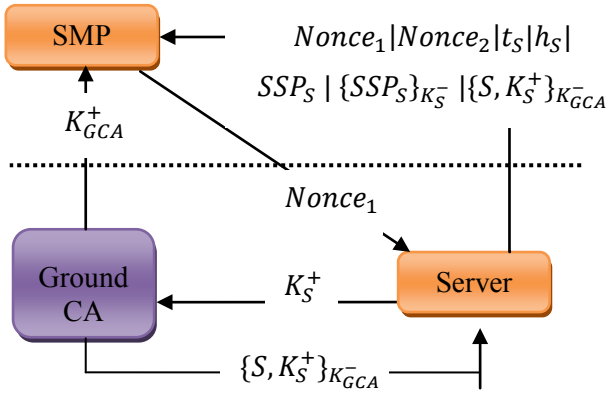

Figure 16. Securing the Negotiation Protocol (Standard PKI Model)

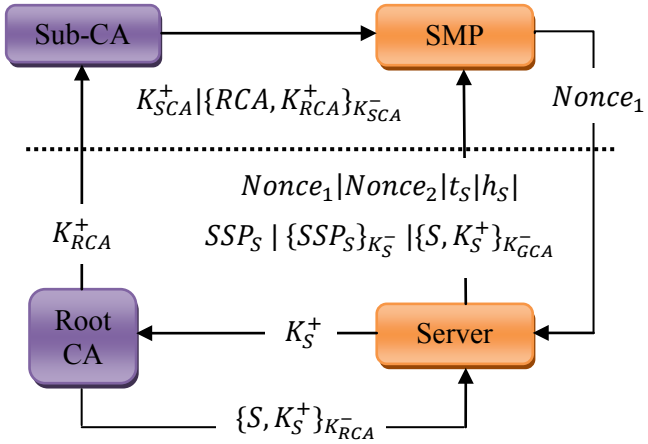

Figure 17. Securing the Negotiation Protocol (Hierarchical PKI Model)

The certification revocation process is not addressed here since we already recommended the used of OCSP and there is no difference between the uses of OCSP for both PKI models (c.f. figure 15). The network cost for the standard PKI model is:

$$N_C.(K_{GCA}^+ + SSP_S \mid \{SSP_S\}_{K_S^-} \mid \{S, K_S^+\}_{K_{GCA}^-} \mid t_S \mid h_S + 2.Nonce_1 + Nonce_2) \cong N_C.(2.l_{sig} + Size_C + 3.Nonce + SSP_S + h_S)$$

The network cost for the hierarchical PKI model is:

$$N_f.K_{RCA}^+ + N_C.(SSP_S \mid \{SSP_S\}_{K_S^-} \mid \{S, K_S^+\}_{K_{RCA}^-} \mid t_S \mid h_S + 2.Nonce_1 + Nonce_2) \cong N_f.l_{sig} + N_C.(l_{sig} + Size_C + 3.Nonce + SSP_S + h_S)$$

Figure 18 shows the network cost comparison between the two models. The hierarchical PKI model is 20% less expensive than the standard model (the average difference data size is about 1408 Bytes).

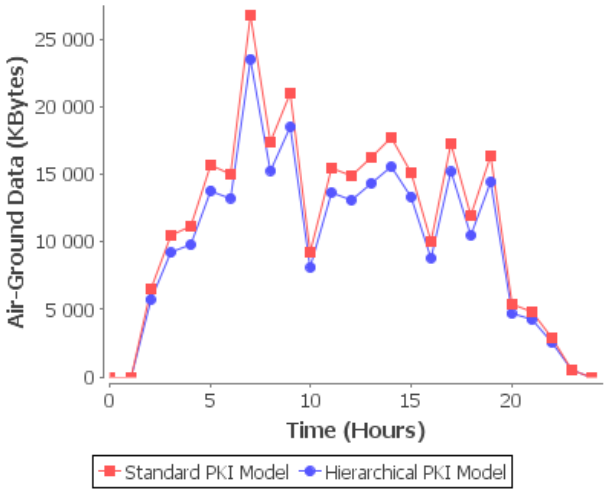

Figure 18. Network Costs to Secure the Negotiation Protocol (Initialization Phase)

Conclusion

In this paper, we presented a new hierarchical PKI model for future ATM systems. We introduced the basic PKI concepts, and then we highlighted the advantages of our model through a performance analysis. We also performed a comparison between the CRL and OCSP revocation approaches.

As the final results have shown, it seems promising to deploy the hierarchical PKI using an online revocation checking status protocol like OCSP. In fact, this combination enhances considerably the network and system consumption performances in an ATM environment. Finally, we used the PKI to secure a negotiation protocol for the supported security mechanisms between two end entities and we quantified the signaling overhead. Again, the hierarchical model performances are better than the classical model.

However, some issues remain unsolved and the study can be extended with some additional features. First, the OCSP server is vulnerable to DoS attacks: when a certificate revocation server is corrupted, end entities (aircrafts, passengers, avionics systems) are enable to check the validity of the certificates and then the integrity of the communications will be compromised. Thus, some modifications are required

to enhance the security of the OCSP server in that way. Also, because of the aircraft's mobility and roaming between two distinct domains, some interoperability problems arise: for instance, when a CA has to manage some aircrafts that do not belong to its domain for instance. Then, the first level of the hierarchical PKI model we proposed has to be investigated to find some solutions to this kind of issues.

Also, the performance study is limited to passengers (as end entities), but it might be interesting to perform some tests for the avionic systems and devices requiring digital certificates for air-ground communications. Also, only the basic version of CRL method and the OCSP protocol have been considered for the revocation scheme comparison: other alternatives such as SCVP or CRL extensions can be added to this comparison study.

References

[1] ARINC, 2007, Draft 1 of ARINC Project Paper 823 Data Link security, Part 1 – ACARS Message Security (AMS).

[2] Richard V. Robinson, Mingyan Li, Scott A. Lintelman, Krishna Sampigethaya, Radha Poovendran, David Von Oheimb, Jens-Uwe Buber, and Jorge Cuellar, 2007, Electronic Distribution of airplane Software and the Impact of Information Security on airplane Safety, The 26th International Conference on Computer Safety, Reliability and Security (SAFECOMP 2007).

[3] Air Transport Association ATA, Revision 2009.1, Aviation Industry Standards for Digital Information Security ATA Spec 42.

[4] EUROCONTROL, 2008, Long-Term Forecast, Flight Movements 2008-2030.

[5] International Air Transport Association (IATA), 2009, World Air Transport Statistics (WATS), 53th edition.

[6] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, June 1999, X.509 Internet Public Key Infrastructure, Online Certificate Status Protocol – OCSP, IETF RFC 2560.

[7] Joel Weise, August 2001, Public Key Infrastructure Overview, Sun Microsystems, Inc.

[8] R.L. Rivest, A. Shamir, and L. Adleman, 1978, A Method for Obtaining Digital Signatures and Public-key Cryptosystems, Communications of the ACM, Vol. 21, Issue 2, Pages 120-126.

[9] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, May 2008, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF RFC 5280.

[10] National Institute of Standards and Technology (NIST), 2002, Federal Information Processing Standards Publication (FIPS) 180-2, Secure Hash Standard.

[11] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer, November 2007, OpenPGP Message Format, IETF RFC 4880.

[12] Phillip Hallam-Baker, 1999, OCSP Extensions, Draft IETF PKIX OCSPX.

[13] T. Freeman, R. Housley, A. Malpani, D. Cooper, and W. Polk, December 2007, Server-based Certificate Validation Protocol (SCVP), IETF RFC 5055.

[14] Dawit Getachew and James H. Griner, 2005, An Elliptic Curve Based Authentication Protocol for Controller-Pilot Data link Communications, International Journal of Computer Science and Network Security.

[15] Richard V. Robinson, Mingyan Li, Scott A. Lintelman, Krishna Sampigethaya, Radha Poovendran, David Von Oheimb, and Jens-Uwe Buber, 2007, Impact of Public Key Enabled Applications on the Operation and Maintenance of Commerical Airplaines, Aviation Technology Integration and Operation (ATIO) Conference, Belfast, Northern Ireland.

[16] Patel, V. and McParland, T., October 2001, Public Key Infrastructure for Air Traffic Management Systems, Digital Avionics Systems, 2001, DASC. The 20th Conference, pages 7A5/1 – 7A5/7 vol.2.

[17] Perlman, R., 1999, An Overview of PKI Trust Models, Network IEEE, Pages 38-43, Vol. 13.

[18] Direction Générale de l'Aviation Civile, Direction du Transport Aérien, 2010, Observatoire de l'Aviation Civile : Tendance et Derniers Résultats du Transport Aérien International.

[19] Ben Mahmoud, MS. and Larrieu, N. and Pirovano, A., 2010, An Adaptive Security Architecture For Future Aircraft Communications, Digital Avionics Systems Conference, 2010, Salt Lake City, USA.

Acknowledgements

We would like to thank Nicolas Staut and Antoine Saltel, students at ENAC for their help and involvement in the performance analysis.

Email Addresses

Mohamed Slim Ben Mahmoud: Slim.ben.mahmoud@recherche.enac.fr

Nicolas Larrieu: Nicolas.Larrieu@enac.fr

Alain Pirovano: Alain.Pirovano@enac.fr