AIAA Aviation Technology Integration, and Operations (ATIO) Conference,
Belfast, Northern Ireland, 18-20 September 2007

Impact of Public Key Enabled Applications on the Operation and Maintenance of Commercial Airplanes

Richard V. Robinson[1], Mingyan Li[1], Scott A. Lintelman[1], Krishna Sampigethaya[2], Radha Poovendran[2], David von Oheimb[3], Jens-Uwe Bußer[3]

[1] Boeing Phantom Works, [2] University of Washington, [3] Siemens Corporate Technology

Abstract. Making airplanes network-enabled can significantly increase the efficiency of aircraft manufacturing, operation and maintenance processes. Yet these benefits cannot be realized without addressing the potential for network-induced security threats. This paper addresses challenges that emerge for network-enabled airplanes that use public key cryptography-based applications. In particular, we focus on the electronic distribution of airplane software and data. We present both an *ad hoc* approach, without trust chains between certificates, and a structured approach employing a PKI. Both approaches facilitate public key-enabled applications, and both levy operational requirements on airlines. We describe the integration of these requirements into existing airline ground infrastructure and processes, to minimize operating overhead. The presented work is based on ongoing collaborative efforts among Boeing, FAA and EASA, to identify needs of the airlines for operating and maintaining network-enabled airplanes.

1 Introduction

The integration of networking capabilities into airplanes enables various applications with benefits ranging from reduced costs to increased passenger comfort and crew convenience. One emerging application replaces the cumbersome and costly transfer of data via floppy disks, CDs, and signed documents, with electronic delivery of software to the airplane and distribution of data such as structural health reports from the airplane [1,2].

However, the safety-critical nature of airplane software and inherent vulnerabilities of data networks together present security threats to airplane safety and airline business, mandating secure solutions [3,4]. The authenticity and integrity of distributed software and data for network-enabled airplanes must be ensured. A solution approach is to employ public key cryptography solutions such as digital signatures [2,5]. While such an approach can enhance the intent of existing FAA guidance for airplane software design and development [6], it imposes unprecedented challenges on airplane manufacturing, operational, and maintenance processes. The requirement to manage cryptographic keys and digital certificates throughout the airplane life cycle is significant among these challenges.

In [3,4] we introduce a generic system for secure electronic distribution of airplane software and data. Following standard terminology as used in the Common Criteria [7], we refer to the pertinent data and software as airplane *assets* and to the various organizations and institutions responsible for handling and protecting those assets as system *entities*. A generic *Airplane Asset Distribution System (AADS)* uses public key cryptography as the basis for asset protection. Using the nomenclature of AADS, this paper focuses on identifying operational requirements for airlines needed to meet the challenges presented by public key-enabled airplane applications. Fig. 1 illustrates the AADS entities and the distribution of assets among them. As seen, an airline may receive software from manufacturers and suppliers, and may pass it on to servicers. Further, during the lifecycle of an airplane, software and data are distributed among the airplane and its manufacturer, owning airline, and/or its servicers.

We assume the use of cryptographically-derived digital signatures to protect the integrity and authenticity of distributed assets in the AADS. The source signs each distributed asset with a *private key*, and appends a *digital certificate* that validates the link between the source and its private key while providing the source's *public key*. The destination uses the certificate to verify the signature on the asset, and therefore must be able to verify, or at least *trust* the certificate. Before we present the

important challenges faced by the airlines in the AADS, we review some background about digital signatures and PKI in general.

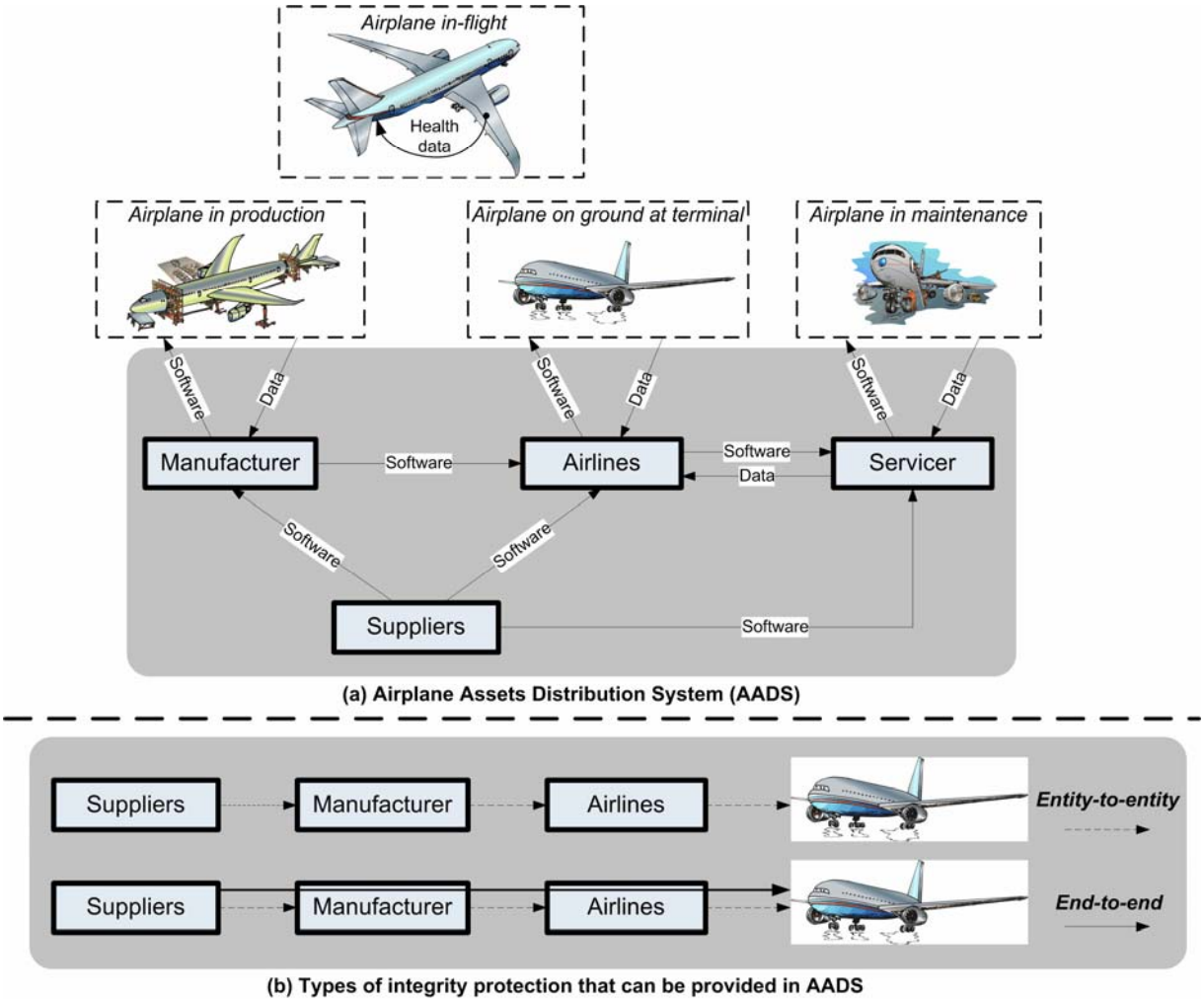

Fig. 1. (a) Distribution of signed assets during lifecycle of the airplane in the Airplane Assets Distribution System (AADS). (b) Illustration of entity-to-entity and end-to-end integrity protection in AADS.

2 Background: Digital Signature and Public Key Infrastructure (PKI)

There are two classes of cryptography: symmetric and asymmetric (or public) key cryptography. In the former, the encryption and decryption keys are the same and must be kept secret. In contrast, public key cryptography employs a pair of distinct keys: a *public key* and a *private key* [8]. The private key is held by a user secretly and the public key may be published. Data encrypted with a public key can be decrypted only with the corresponding private key and vice versa. Digital signatures exploit this as follows: A sender digitally signs data, such as electronic documents, by applying a cryptographic operation, involving its private key, on a digest of the data. The resulting signature is attached with the data and sent to the receiver. The receiver obtains the digital certificate of the sender, which provides the sender's public key and confirmation of the link between its identity and the public key. The signer's certificate is often attached to the signed data. The receiver either trusts this certificate or must be able to verify it. The signature on the data can then be verified using the public key contained in the certificate. This verification ensures the authenticity and integrity of the received data.

Whenever a large group of entities must communicate securely without necessarily knowing or trusting each other directly, a public key infrastructure (PKI) may be used to contain the cost of

deploying and managing digital certificates. A PKI typically consists of a *Registration Authority (RA)* that verifies the identities of entities, authorizes their certificate requests, and generates unique asymmetric key pairs (unless the users' certificate requests already contain public keys); and a *Certification Authority (CA)* that issues corresponding digital certificates for the requesting entities. Optionally, a *Certificate Repository* may be present, which stores and distributes certificates and/or a certificate revocation list (CRL) identifying the certificates that have been declared invalid. Each entity possessing the authentic public key of a CA can verify the certificates issued by that CA.

As an alternative of using a PKI, the public key of an entity can be distributed directly to all potential signature verifiers, so long as the key's integrity is protected by physical or other means. This is often done using the format of a self signed certificate as a container for the public key and the corresponding owner identity. Certificates of this type are referred to as trusted certificates in the following sections. Such certificates cannot be verified. In order to remain trusted, they must be stored with write access control or other means of integrity protection.

3 Overview of Challenges and Operational Requirements for Airlines

Establishment of trust among AADS entities. As seen in Fig. 1(a), airlines receive software from several other entities and must be able to verify signatures provided by each of them. This requires trust to be established between the entities in the system.

Entity-to-entity and end-to-end integrity protection in AADS. As illustrated in Fig. 1(b), the integrity of software and data can be protected using either of two schemes. In an *entity-to-entity* protection arrangement, the airline verifies signed software from suppliers, re-signs and distributes to airplane. This contrasts with an *end-to-end* arrangement in which supplier-signed software is verified by the airplane. Entity-to-entity integrity protection requires the airplane to trust certificates of airlines only, and benefits from the *ad hoc* solution described in Section 4. On the other hand, end-to-end integrity requires the airplane to possess trusted or verifiable certificates of all the suppliers, and is thus more suited for the structured solutions described in Section 5.

Number of certificates managed by airlines. The registered entities that require certificates include the suppliers, the manufacturer, the network-enabled airplane fleet and the airline servicers or personnel signing and verifying the airplane assets. Therefore, the expected number of certificates depends on the scale of the airline operation, including the fleet size and number of users/servicers. It is desirable to keep the number of certificates low. As will be seen in Section 4, the number of certificates can be a major concern when using the *ad hoc* solution for end-to-end integrity protection.

Assurance requirements of AADS. It is shown in [4] that to adequately meet security threats to safety-critical assets, such as RTCA/DO-178B [6] Level A software parts, AADS must be assured at a Common Criteria [7] Evaluation Assurance Level (EAL) of at least 4 (methodically designed, tested, and reviewed). To the degree that reliance upon certificate integrity supports assurance of airplane software part integrity, the creation, issuance, distribution, and verification of certificates must also be assured at a minimum of EAL4.

Certificate requirements of AADS. In order to support digital signatures in AADS, the airline may choose to adopt the common X.509 format. In order to assist verifying that the airplane assets are signed by authorized entities, certificates issued to authorized entities can have an exclusive usage type to distinguish them from certificates issued by the same authority for purposes and applications other than AADS. For example, the X.509 v3 standard certificates allow certain attributes to distinguish their service for AADS. Further, the airlines must choose a strong signature algorithm. In July 2007, e.g. RSA 2048 bit with SHA-256 is considered to be robust against cryptanalytic attacks, but advances in cryptanalysis and computing power may change this soon. The lifetime of signatures, in particular those used in certificates, is another issue that must be addressed by the airlines.

Signing and verifying assets. The airlines, as well as the airplanes, must be capable of verifying signatures and certificates on received assets, and be capable of signing assets before distributing them.

Registration of signing entities at airlines. The suppliers, airplanes and airlines servicers and/or personnel signing the assets must be identified by the airline before certificates are issued for them. Further, role assignments may be more detailed to specify which entity is entitled to sign which type of asset, e.g. the safety-critical, RTCA/DO-178B level A software parts.

Key and certificate distribution to airplanes. The distribution of keys and trusted certificates to requesting entities, especially to airplanes, is challenging. This distribution must be protected either by secure online or out-of-band mechanisms. A good approach is to enable onboard generation of the airplane private key and to ensure authentic distribution of the corresponding certificate request to the airline CA, or authentic distribution of the airplane's self-signed certificate to all off-board components which require it for authenticity verification of status reports. Similarly, the integrity and authenticity of trusted certificates (either airline certificates or the CA root certificate) must be protected during distribution to the onboard repository.

Key and certificate storage onboard airplanes. The security of the AADS depends on the integrity and authenticity of trusted certificates and the confidentiality of private keys. Further, an airplane may interface with multiple networks, and its Line Replaceable Units (LRUs) – some of which store keys – may be replaced as needed over its lifecycle. Such unique constraints require careful consideration to protect the airplane's private key.

Revoking certificates onboard airplanes. In the case that a ground system's private key becomes compromised, each certificate of that ground system must be revoked in order to prevent any misuse of signatures. Consequently, before verifying a signature, the certificate status must be validated by the airplane. Similar arguments hold for the compromise of an airplane key, upon which the airplane certificate must be revoked. Note that expired certificates are considered invalid, and therefore need not be listed in the CRL.

Fallback mechanism at the airlines. The airline must be capable of using a fallback system, e.g. transfer of software parts on CDs, in the event of failure or unavailability of any AADS component.

Recording and auditing at the airlines. All security-related events throughout the AADS must be logged, and event logs protected against tampering. Further, for traceability, the reason for certificate revocation must be recorded and outdated certificates must be archived for a period of time.

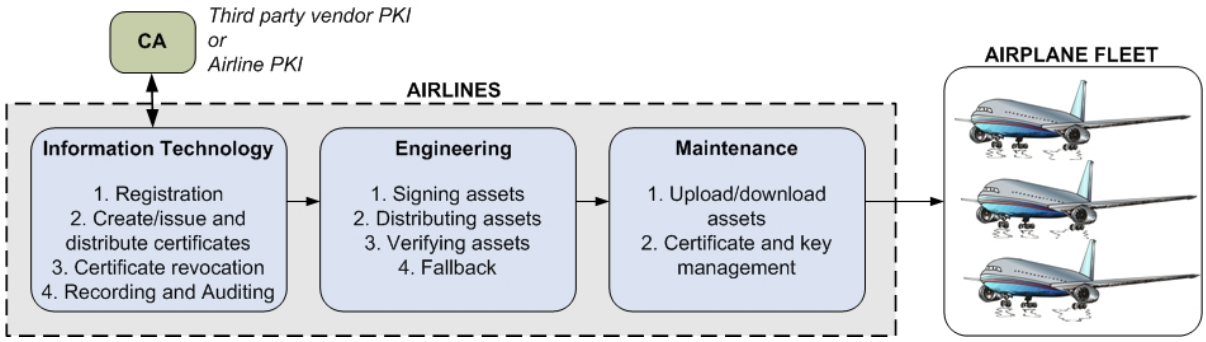

Fig. 2. An assignment of new operational requirements across typical departments at airlines

Fig. 2 illustrates a possible allocation of the above operational requirements imposed on the airlines by the AADS. The Certificate Authority (CA) may be an offline third party vendor from which certificates are purchased, or an online entity external or internal to airlines. To minimize overhead, these requirements may be integrated into the typical infrastructure (departments) and processes existing at many airlines. The airlines may have different approaches to handle certificates. Based on the approach, there can be an *ad hoc* and a structured solution. We present these solutions below.

4 *Ad Hoc* Solution: Preloading Trusted Certificates

In this approach, no central authority certifies all public keys. Instead, trusted collections of public keys are contained in certificates. The certificates may be self-signed, or signed by a local certificate authority, or obtained from any third party. The validity of the certificates themselves can not be verified by the airplane. Therefore the airplane must preload them via a trusted out-of-band mechanism. The same holds for the airplane certificate used to downlink data to ground systems. Further, all certificate stores must be updated periodically in order to maintain the status of certificates and to implement revocation.

Advantages. The main advantage of this solution is that it is simple. It does not need additional infrastructure but can be improvised rather easily, and a high-assurance security evaluation, for instance at EAL6 (semiformally verified design and tested), could be done without incurring overwhelming costs. Having a limited number of certificates to check and trust, and a limited number of corresponding private keys, reduces the probability of having a compromised key or an invalid certificate, under the assumption that a private key is not shared among multiple entities. The need to distribute only a limited number of certificates is good from a security point of view.

Drawbacks. The solution does not scale for end-to-end integrity protection, since in this case the airplane requires certificates of all suppliers apart from the airlines and manufacturer to be preloaded, as can be seen in Fig. 3 (where the required certificates are indicated in grey). Further, each airline must be able to obtain certificates from all its suppliers via an authentic channel. As the scale of the airline increases, certificate management is presented with more challenges due to increased dynamics in the ground infrastructure and increased fleet size.

5 Structured Solutions: Use of PKI

A better, long-term solution is to make use of a full *Public Key Infrastructure (PKI)* [8]. The airline can assign the role of CA and/or RA either to a trusted third party, e.g. a commercial vendor or a Federal agency, or can implement its own PKI and itself function as CA and RA. The choice is governed by several factors, including deployment and operational costs, resource considerations, number of certificates, and trust established with third party vendors [8].

Advantages. A structured PKI solution offers long-term practical benefits in terms of scalability and flexibility. As illustrated in Fig. 4, only the CA self-signed root certificate (marked grey) needs to be distributed authentically. Moreover, the availability of online certificate status checking or certificate revocation lists (discussed below) for certificate status verification by airplane enables a more secure approach than assuming correct status for all preloaded trusted certificates.

Drawbacks. The structured solution is relatively more expensive than the *ad hoc* solution, incurring costs in set up and maintenance of PKI and its functionalities. Additionally, since the PKI is a crucial security mechanism of the AADS, it also must be evaluated (at the same assurance level), which means a significant extension of the scope of the AADS evaluation. So the resulting stringent technical and non-technical requirements on the PKI implementation may increase the costs.

5.1 Certificate Revocation

To cope with changes in business relationships or airline internal authorization rights, and for the case of compromise of a private key, certificate revocation has to be established. Commonly used solutions to obtain the revocation status of certificates are:

5.1.1 Online Certificate Status Protocol (OCSP)

If online connectivity is established, the airplane can check the validity status of a certificate by sending a status request to the corresponding CA, containing a cryptographic fingerprint of the certificate. The CA checks whether the certificate was revoked, and replies to the requester a signed status report. For retrieving the revocation status of X.509 certificates, the standardized protocol OCSP [9] can be used.

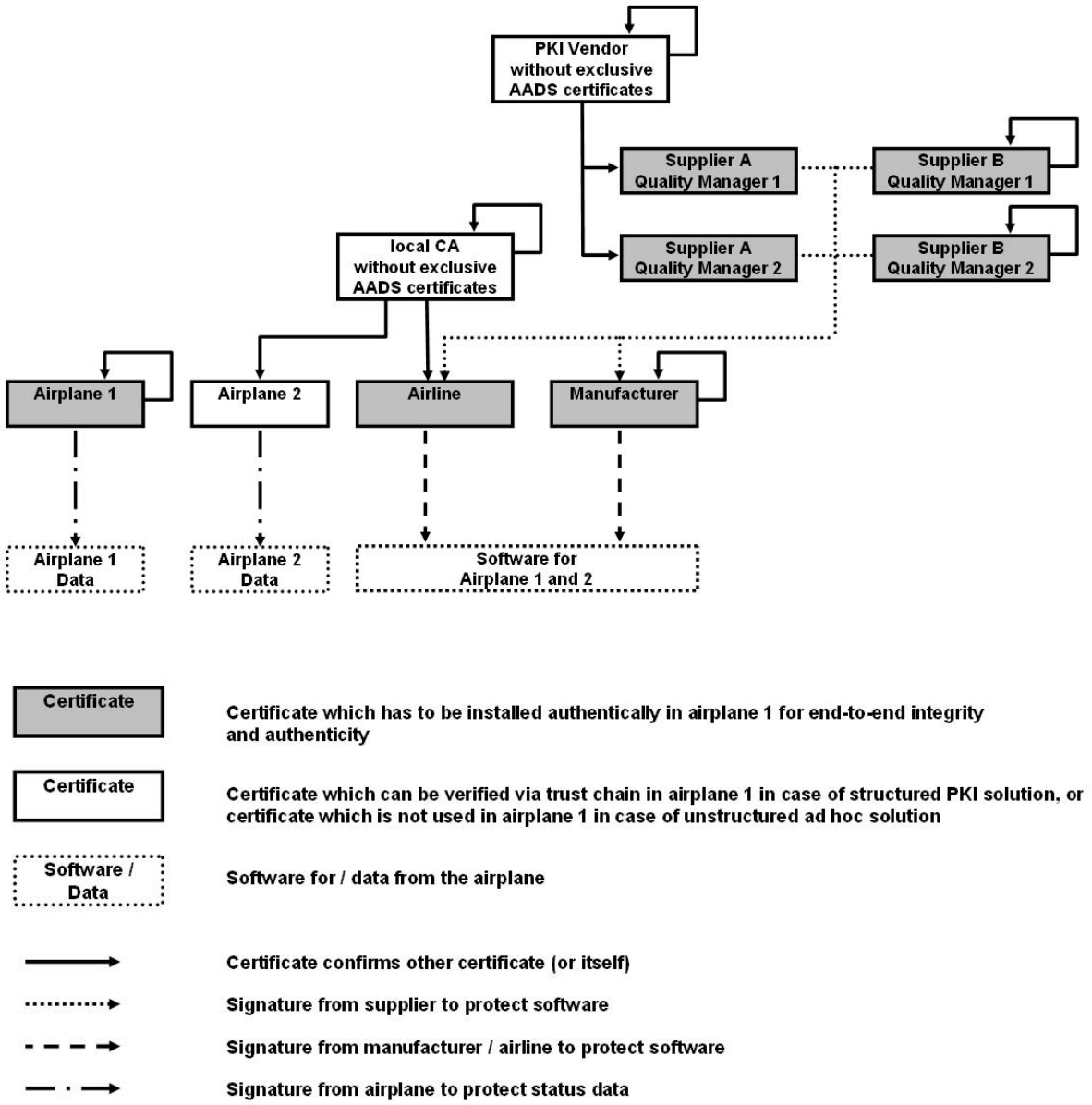

Fig. 3. Ad hoc solution for providing trusted certificates. External PKIs (like shown for supplier A and airline) or self-signed certificates (like shown for supplier B and the manufacturer) may be used in this scenario. The airplane has its own certificate (e.g. a self-signed one created on board) to protect data sent to airlines (slash-dotted line). Suppliers sign the airplane software to protect integrity (dotted line); but this signature may be removed at manufacturer or at airlines because it cannot be checked in the airplane anyway (except the airplane gets the certificates from all the suppliers; marked grey), and replaced with new signatures (slashed line). Quality managers are considered to be entities approving, releasing and signing software at suppliers.

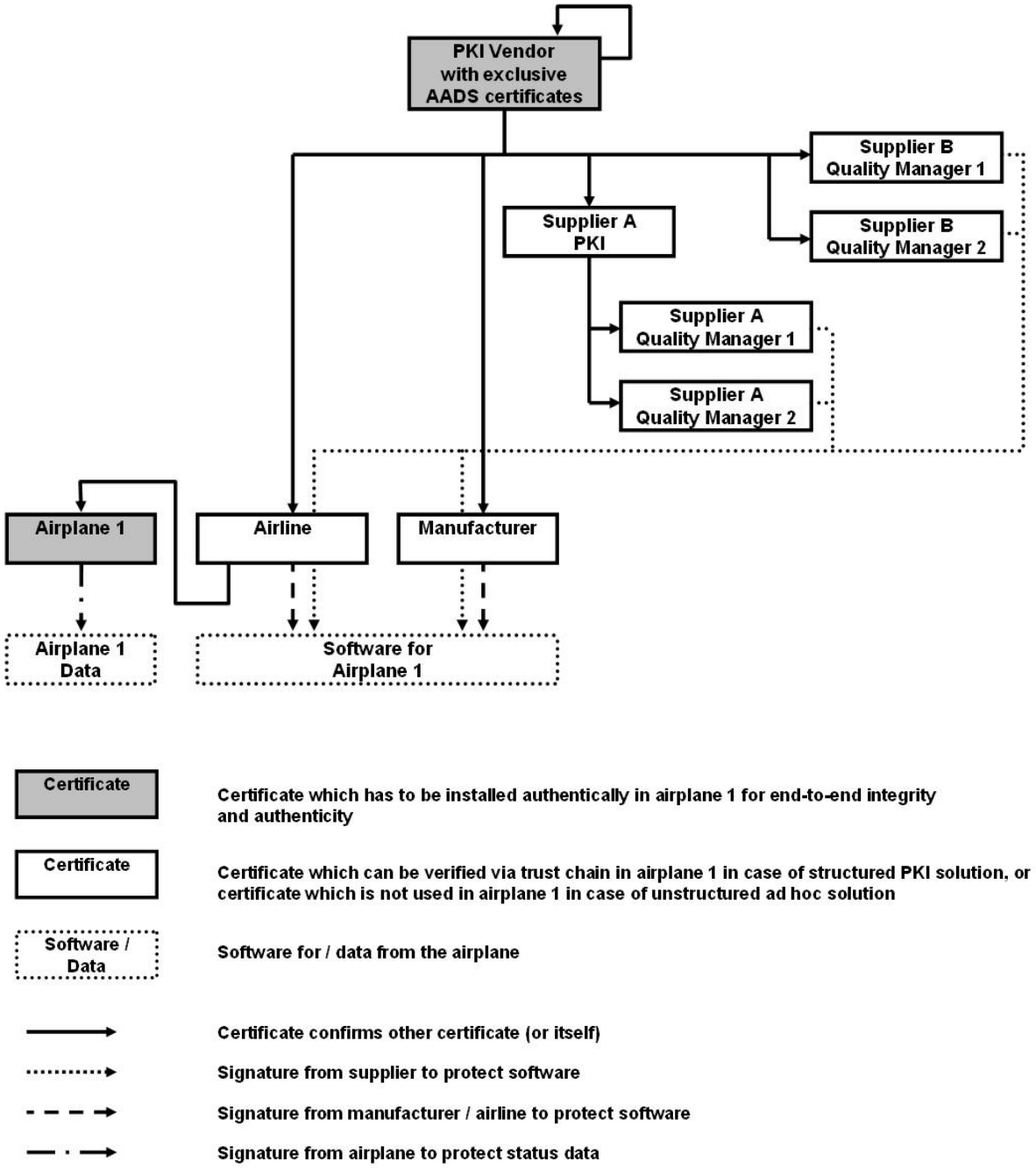

Fig. 4. Structured solution using PKI. The signature from the supplier (dotted line) stays on the software distributed not only to manufacturer and/or airlines, but also to the airplane, hence achieving end-to-end security. The airplane requires only the self-signed certificate of the PKI vendor apart from its own certificate (marked grey). All other certificates required for checking trust chains need not be pre-loaded on the airplane, but may be sent later with the signed software. Quality managers are considered to be entities signing software at supplier.

Advantages. This method allows real-time checking of certificate status. Only the certificate of the status server is required to be distributed authentically. Different CAs can be used in parallel, but then several status server certificates have to be distributed, of course. Sub-CAs can be used even without distributing further certificates because they are certified by the root CA.

Drawbacks. Direct online connectivity to the CA or a status server is required whenever a certificate is used; that means connecting the airplane to the airline's company network, and maybe even to the Internet if supplier certificates have to be checked, too, and therefore making potential security vulnerabilities accessible much more often. If the connection cannot be established, the use of certificates is delayed or blocked. Much higher CPU performance and bandwidth of certificate repository servers are required, and not all CAs support this method.

5.1.2 Certificate Revocation Lists (CRL)

The CA creates lists of (serial numbers of) revoked certificates (CRLs) with a short validity time (e.g. a day or a week), and signs these lists. The CA creates a new CRL every time a certificate is revoked or at latest short before the last CRL becomes invalid. A requester can download the latest valid version of the CRL from a server and can use the CRL offline to check the validity status of a certificate. The CRL can be used at the requester as long as it is valid, however when connectivity is given, the requester can look for an up-to-date CRL every time when checking a certificate status.

Advantages. No direct connection to the CA or a status server is required. The CA has to compile and to offer for download just one list for all users, instead of checking the status of many certificates, so the performance requirements for the CA or CRL server are much lower. This solution is most commonly used.

Drawbacks. The information about the certificate's status is not as "fresh" as in the case of the online status retrieval, though the difference is marginal in practice.

Table 1 summarizes the benefits and drawbacks of the presented solutions to providing trustworthy certificates to the airlines.

	Ad hoc Solution	**Structured Solution with online protocol**	**Structured Solution with CRL**
Benefit	Short-term	Long-term	Long-term
Effort for high-assurance evaluation	High	Very high	Very high
Scalability	Low	High	High
Certificate validity check	None (only manual revocation possible)	Real-time (online connection required)	Not real-time (offline possible, but up-to-date CRL required)

Table 1. Comparison of ad hoc and structured solution

5.2 Managing Multiple CAs

Some suppliers, manufacturers, airlines, and servicers may already have different PKI solutions with their own local CAs within their companies. Therefore, it seems not practical that they all will agree to use the same PKI provider, but instead they would like to connect their solutions to form a common trust network.

Local CAs can be connected via:

1. **Common root CA:** The (root) public key of each CA is signed by a new CA (Fig. 5a). This solution is the logically most simple and straight-forward solution. Yet it has the disadvantage that the root certificate changes which is installed as trust anchor in many entities and devices. This can cause a lot of organizational effort. Furthermore, the local CAs give away the control to another instance.
2. **Cross certification**: Each CA certifies the (root) key of all other CAs which it trusts (Fig. 5b). This solution avoids the disadvantages of changing root certificates and giving away control, but may cause a lot of effort if there are many CAs which need to be brought together.

3. **Bridge CA**: Each CA certifies the public keys of the bridge CA, and gets its public key certified by the bridge CA in return (Fig. 5c). Each CA has to certify only one other CA (namely the bridge CA), and to be certified by this, which reduces the effort strongly compared to solution 2. It may be seen as a mixture of solution 1 and 2.
4. **Trusted lists**: The entities maintain lists of trustworthy CAs, so they have several root CAs. This solution is similar to the ad hoc solution discussion in section 4, but on a higher hierarchy level (not shown in Fig. 5).

A mixture of different connection methods is possible, that means e.g. some CAs may agree to be connected using a common root CA, whereas the connection to other CAs is established via cross certification or a bridge CA.

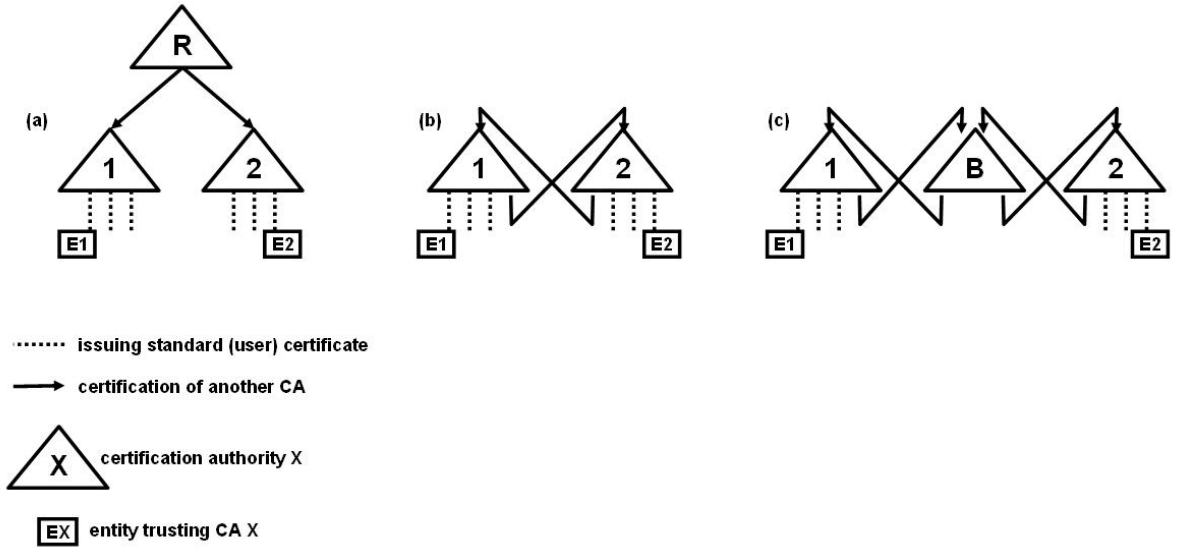

Fig. 5. Methods for connecting separated PKIs to a common PKI: (a) A new CA certifies the existing CAs 1 and 2, and becomes the new root CA (trust anchor) for all entities (e.g. E1 and E2) which trusted CA 1 or CA 2 before. (b) Cross certification means that CA 1 certifies CA 2, and vice versa. So CA 1 is still the root CA for entity E1, and CA 2 is root CA for E2. E1 sees CA 2 as trusted sub-CA of CA 1, and therefore E1 trusts E2. (c) CA 1 certifies bridge CA B, and is certified by B; CA 2 does the same. CA 1 stays the root CA for E1, but because B is a trusted sub-CA of CA 1 and therefore CA 2 is a trusted sub-sub-CA of CA 1, E1 trusts E2.

6 Conclusions

We have introduced and discussed several solutions for managing public keys required within an Airplane Assets Distribution System.

The unstructured ad hoc solution may only be good for a first rollout of an AADS. For long-term operation, a PKI solution seems much more appropriate, although the evaluation and deployment efforts are higher. Furthermore, we recommend supporting different local solutions for different companies, and using a commercial or federal CA as root or bridge CA for the inter-company communication between suppliers, manufacturers, and airlines. Revocation mechanisms have to be provided. We suggest using CRLs for checking certificates at the airplane to avoid the necessity of direct connectivity to external networks.

Acknowledgements

We thank Prof. Peter Hartmann from the Landshut University of Applied Sciences for his insightful discussion that helped us to improve this paper.

References

[1] ARINC Report 665: Loadable Software Standards, ARINC, August 2005.

[2] G. Bird, M. Christensen, D. Lutz, P. Scandura, Use of integrated vehicle health management in the field of commercial aviation, NASA ISHEM Forum, 2005.

[3] S. Lintelman, R. Robinson, M. Li, D. von Oheimb, K. Sampigethaya, R. Poovendran, Security Assurance for IT Infrastructure Supporting Airplane Production, Maintenance, and Operation, National Workshop on Aviation Software Systems, Available online: http://chess.eecs.berkeley.edu/hcssas/papers/Lintelman-HCSS-Boeing-Position_092906_ 2.pdf

[4] R. Robinson, M. Li, S. Lintelman, K. Sampigethaya, R. Poovendran, D. von Oheimb, Challenges for IT Infrastructure Supporting Secure Network-Enabled Commercial Airplane Operations, in AIAA Infotech@Aerospace Conference, 2007.

[5] ARINC Electronic Distribution of Software (EDS) Working Group. http://www.arinc.com/aeec/projects/eds/

[6] DO-178B: Software Considerations in Airborne Systems and Equipment Certification, Radio Technical Commission for Aeronautics (RTCA), December 1, 1992.

[7] Common Criteria. http://www.commoncriteriaportal.org/

[8] Understanding PKI: Concepts, Standards, and Deployment Considerations, C. Adams and S. Lloyd, 2nd edition, Addison-Wesley, 2003.

[9] IETF Draft 2560: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP, 1999. http://tools.ietf.org/html/rfc2560