An adaptive security architecture for future aircraft communications

Mohamed-Slim Ben Mahmoud, Nicolas Larrieu, Alain Pirovano, Antoine Varet

AN ADAPTIVE SECURITY ARCHITECTURE FOR FUTURE AIRCRAFT COMMUNICATIONS

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

Abstract

This paper presents an original and adaptive security architecture for the future connected aircrafts. A secure system topology for the embedded network is proposed with regards to network and system constraints, service priorities and regulatory recommendations. The design of a new component called Security Manager (*SecMan*) is explained in details and all its processes are formalized for a better understanding of the proposal made in this paper. A performance study is done in order to assess the advantages of this adaptive security policy within some critical aircraft communication scenarios.

The adaptive security architecture will be applied for a satellite-based system architecture of an industrial project titled "*FAST*" (Fiber-like Aircraft Satellite Telecommunications). The project is 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 *FAST* satellite system aims at providing bi-directional satellite communication services on commercial aircraft worldwide. Many partners take part in the project, both from industry (EADS Astrium, Axess Europe, Vodea, and Medes) and academy (CNRS/LAAS, ISAE, ENAC, Telecom Bretagne).

Introduction and Problem Statement

New Aeronautical Applications and Telecommunication Networks: Future Trends

The means of communication for aviation is evolving rapidly in line with progress in new technologies. Nowadays, the voice still remains the main vehicle for air-ground communications, using either Very High Frequencies (VHF) or HF bands, but the industry is expected to see a transition from voice to data communications in the near future.

Nowadays, data link systems, such as the Controller Pilot Data Link Communications (CPDLC) are being used for many reasons. First, when a voice radio communication is used, all pilots being assigned to a single air traffic controller are tuned to the same frequency. This is a major issue considering the increasing number of flights and the air traffic controller workload: the European Organization for the Safety of Air Navigation (EUROCONTROL) reported in the 2009 Network Operations Report [1] an increase of 14.77% of the average daily traffic between 2004 and 2008 in Europe. Long term forecast studies [2] reported an average air traffic growth of 2.3% - 3.5% per year between 2007 and 2030. Frequencies can be saturated when assigned to many pilots, communications can supersede each others, transmissions may have to be repeated if the voice message is not clear and communication errors increase.

Previously, aeronautical organizations proposed to divide air traffic sectors to make easier the Air Traffic Management (ATM), but this solution seems to have many feedbacks. First, the number of available voice communications channels is limited: there may not be new channels available when reaching a saturation point in high density airspace for instance. Also, sector division requires transferring a flight between two successive sectors: when the aircraft is crossing the sector borderline, the controllers assigned to each sector have to coordinate for the success of the handover procedure.

The data link communications offers a more efficient conflict resolution strategy in order to increase the effective capacity of the communications channel, reduce delay in severe weather, and enhance pilots and air traffic controller working environment. In [3], interesting statistics about CPDLC versus voice radio communications can be found. The authors showed that the reduction in analog voice usage increase linearly as a function of CPDLC equipage: for 100% CPDLC aircraft equipped, the radio voice usage decreases by 84%.

Using data link communications increases the development of data-based services. According to the

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

Communications Operating Concept and Requirements for the Future Radio System (COCR) document [4], jointly produced by EUROCONTROL and the Federal Aviation Administration (FAA), new data communication services will emerge around 2020 and replace voice progressively to become the dominant mean of air-ground communications.

Besides classical services (Air Traffic Services – ATS and Aeronautical Operational Control Services – AOC), airlines will probably provide new passenger dedicated services (Aeronautical Passenger Communication Services – APC) such as broadband Internet access, or new safety-related services. Additional applications may be supported in a long term: recent aeronautical projects (such as *FAST* project we introduce in this paper) are considering new operational services such as medical supervision and video surveillance systems. A medical supervision system is a medical device used in flight during a medical emergency to transmit live medical information to ground medical staff. Video surveillance system is a monitoring terminal connected to a set of cameras dispatched through the cabin and the cockpit to prevent malicious and suspicious behaviors.

On the other hand, enhanced technologies (e.g. SATCOM) offer new link capacities and allow us to aggregate all these different kind of traffics on a single link. Up to now, each class of services has its own communication infrastructure, but it seems favorable to integrate them all on the same system for cost-saving purposes. Recent researches have shown that the convergence to an aggregated air-ground data-based communication system is a trend for future aircrafts: [5] studied the feasibility of an hybrid aeronautical system where ATS, AOC, and APC traffics are mixed on the same satellite link.

The heterogeneity of the global aggregated traffic and a seamless inter-operation of such systems with existing ground networks seem to indicate a need for an IP-based aeronautical network. A new generation of IP-based data links using satellite broadband communications is expected to merge. Current aeronautical networking standards are base on the ISO OSI protocol suite and are referred to by the International Civil Aviation Organization (ICAO) as Aeronautical Telecommunications Network ATN/OSI. But ICAO has also defined an ATN/IPS based on the Internet Protocol Suite (IPS) to enable the use of Commercial-Off-The-Shelf (COTS) technologies [6], mainly to reduce the overall system development and maintenance costs.

Security Issues for Future Aeronautical Networks

The impact of these meaningful changes is a substantial increase in security risk in a critical context where human lives are involved and breakdowns in air-ground connectivity cannot be tolerated. Therefore, security has become a major concern in the civil aviation industry. Security mechanisms cannot be deployed in an indiscriminate manner: regulatory recommendations prescribe the separation of ATS traffic from APC and AOC traffics, then the network topology has to be adapted to such guidance and priority techniques must be used. ATS and AOC communications are less demanding in terms of data rate than APC, but they are much more stringent in terms of availability and reliability as they are dealing with flight security and safety.

If COTS products are used, the overall system security has to be watched carefully: when technologies are public and their specifications and developments are out of control of the aeronautical community, new discovered vulnerabilities and threats using these products have to be followed and repaired as soon as possible. Also, Quality of Service (QoS) requirements, constricted bandwidth of the air to ground link, and weather conditions (which impacts the transmission quality) impose the reduction of additional overheads: any “static” security policy will be expensive in this case.

Traditionally, QoS and security issues have been treated separately. Clearly, this cannot be the best choice to make: security mechanisms may severely affect network or system performances in a context where the resources are limited. On the other hand, the applied security level for each application strongly depends on available network or system resources. Then, security and QoS have mutual dependent performances, and a trade-off between environment performances and security policies seems to be useful and quite relevant for the studied context.

This paper aims to address all these issues through an adaptive security architecture for future aeronautical communications. First, the system

architecture proposed for the *FAST* project is introduced. Then, the security architecture's core is presented (*SecMan* decision making module): the algorithm is presented in a formal manner. Finally, some validation and performance tests are discussed to highlight dynamic and adaptive *SecMan* features.

Related work

Security for digital aeronautical communications has been considered in few works. The Aeronautical Radio Inc. (ARINC) transport communications provider introduced the Aircraft Communications Addressing and Reporting System (ACARS) to allow data messages exchange between the air and the ground. Given the fact that many eavesdropping attacks have been done on the system, a new and secure version was proposed to avoid this kind of issues (ACARS Message Security – AMS) [7]. AMS is a good security framework to secure ACARS message but the technology became obsolete as soon as ATN /IPS has been defined as the future standard for ATS and AOC data-based services.

[7] investigated an elliptic curve based authentication protocol for CPDLC. Mutual authentication between aircraft and the ATS ground systems are then provided to avoid masquerading and spoofing attacks. As for [8], the security framework is efficient only for cockpit communications: cabin communications and service priorities have been ignored. These are key factors for the future connected aircrafts, and then must be considered in any security analysis. In [9], only recommendations of use are proposed to enhance the security of the air-ground communications using COTS products and an overview and the ATN security concept is introduced (architecture, security requirements, and security framework).

Many security gaps in several satellite-based technologies exist, especially when IP networks are considered. [10, 11] analyze a set of threats and security requirements for IP-based satellite, it has been shown that a layer-2 security is not adapted and a layer-3 security framework has been proposed [12]. In [13], we presented a state of the art paper where aeronautical security data link activities and used mechanisms are reviewed. Advantages and drawbacks of each security solutions are discussed then an adaptive satellite-based security architecture was proposed. Basically, the work described below is an extension of the idea introduced in [13].

Satellite-Based System Architecture: Network Topology

Figure 1 gives an overview of the secure system architecture proposed for the *FAST* project. The system comprises an aircraft on-board segment, a satellite segment, and a ground segment. The ground system is formed by a satellite gateway (GW) connected to an ATN router for aeronautical services, and an Internet router for passengers services. On-board, a satellite terminal is connected to an ATN router for the ATS traffic and a New Generation (NG) router connected to the APC, AOC, medical supervision and video surveillance systems (noted AOC NG – Next Generation).

APC and medical supervision system are connected to Wireless Access Points (WAPs) dispatched across the aircraft. The Digital Video Broadcasting (DVB) standards are used as access layer network in the system architecture: the selected air interface is based on DVB-S2 [14] standard on the forward link, and on the well-adapted for mobile applications DVB-RCS [15] standard for the return link. Two *SecMan* Proxies (SMPs) are connected to the NG and ATS routers and located into Demilitarized Zones (DMZs) using advanced firewall features. Request connections are redirected to the SMP where SecMan treats individually each request and establishes an adapted security policy (see next section for selection mechanism details).

SMP_1 takes into account many type of traffic (APC, AOC, and AOC NG), so it uses both the priorities and the environment state information (both network and system parameters are considered) as inputs to define the security policy (we call it the *Intra-class operational mode*). On the other side, SMP_2 considers only ATS traffic, so only the network and system state information are considered to define the security policy (*Inter-class operational mode*).

A two-level QoS policy is defined to manage the priorities between the services and to allocate the network resources. However, the QoS management is out-of-scope of this paper; more details about the used resources management techniques for this architecture can be found in [16].

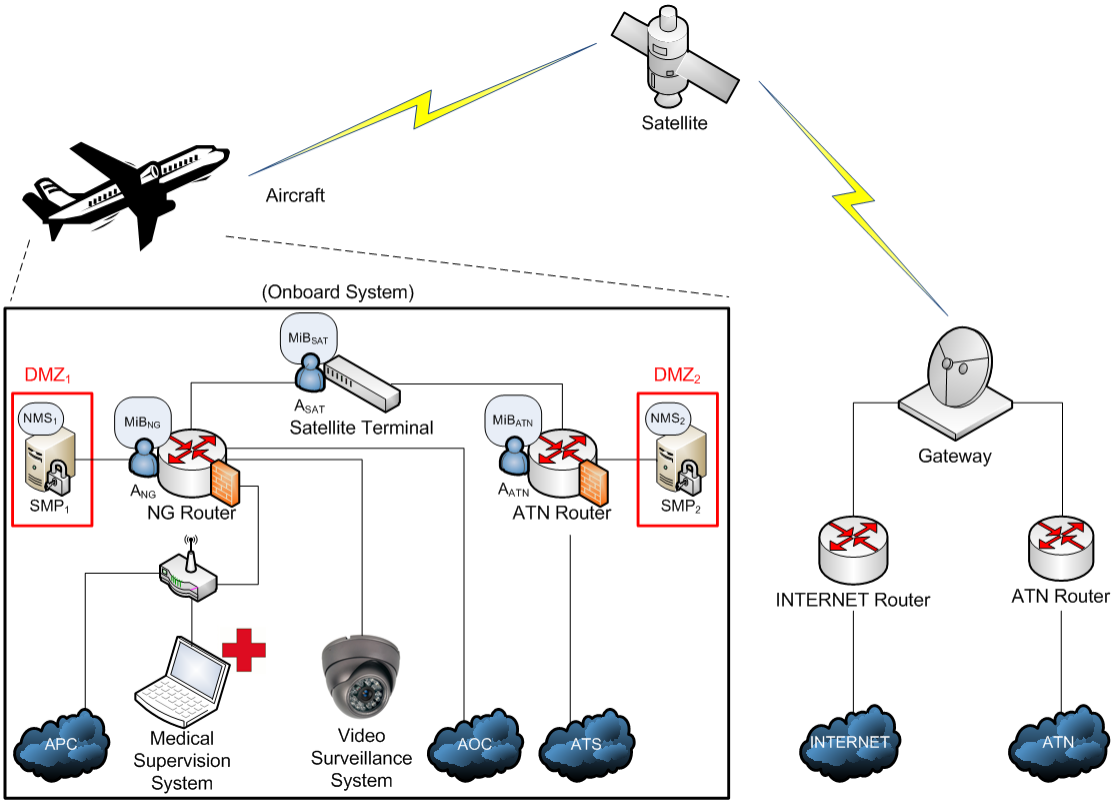

Figure 1. Security Architecture for Satellite-Based Aeronautical Digital Communications

The proposed architecture is designed with respect to:

- The European Telecommunications Standards Institute (ETSI) Broadband Satellite Multimedia (BSM) architecture for IP-based satellite links: segregation between Satellite Dependent (SD) and Satellite Independent (SI) layers is provided as recommended in [17];
- ICAO Standards and Recommended Practices (SARPs): the separation between ATS and the rest of the global traffic is mandatory [18].

SecMan Framework

In this section, we present more precisely the adaptive security management proposal with a particular emphasis on the *SecMan* module: the protocol architecture and algorithm components are detailed.

Protocol Architecture

Figure 2 depicts the general *SecMan* design principle:

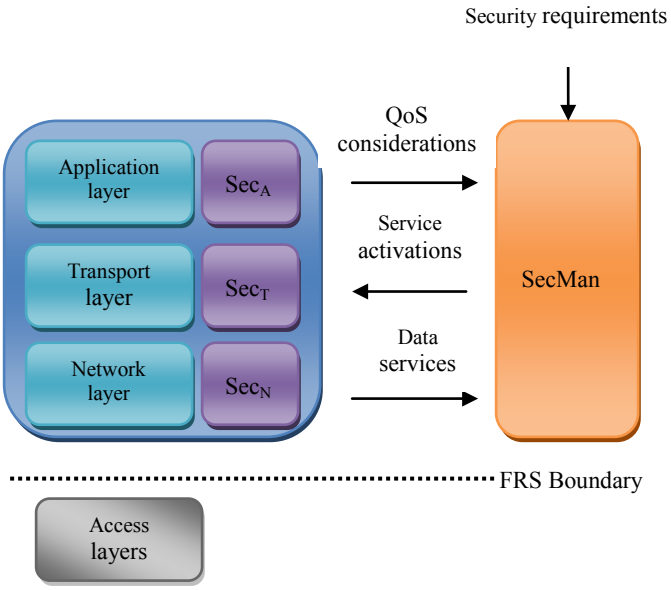

Figure 2. SecMan Protocol Architecture

The module is intended to work above the Future Radio System (FRS) [4] boundary on the control plan. Thus, *SecMan* is technology-independent and compatible with any access network.

As we can see, *SecMan* inputs are:

- *The data services*: every data stream is sent with the priority of the service (ATS, AOC, AOC "NG", or APC);
- *The security requirements*: every service has its own minimum security requirements (for every security service);
- *QoS considerations*: these are the environment information (for instance network and system available resources) which is sent to *SecMan* when a security policy has to be established.

SecMan uses a pre-defined database which contains the alternatives (basically the security algorithms) to define the security policies. This set of mechanisms (called SSPD – Supported Security Protocols Database) is the list of the security protocols (with their parameters) negotiated between the on-board SMP's and the ground entities, as a server for instance (see next section for more details).

SecMan uses a Multi-Criteria Decision Making algorithm (MCDMA) [19] to establish a ranking among the supported security mechanisms and select the best security policy. The purpose of using an MCDMA is to provide a decision making support when many criteria are considered in the system: the Analytical Hierarchical Process (AHP) [20] is the MCDMA approach used in our adaptive security management. AHP methodology, criteria and alternatives are detailed later in this section.

The aim is to define the best security policy for every connection request. By best, we do not necessarily mean the strongest security level, but the highest trade-off between security level and network (and system) cost. In the last section of the paper, the balance between security policy level and security effects on system performances is discussed through the preliminary results we have recently obtained.

The policy can be one protocol-based, but also a hybrid security. Indeed, most of related works have explored security protocols individually without investigating possibilities and performance improvements associated with the integration of a multi-layer security policy.

SecMan Algorithm Design

The flow chart of Figure 3 gives a closer look to the *SecMan* algorithm:

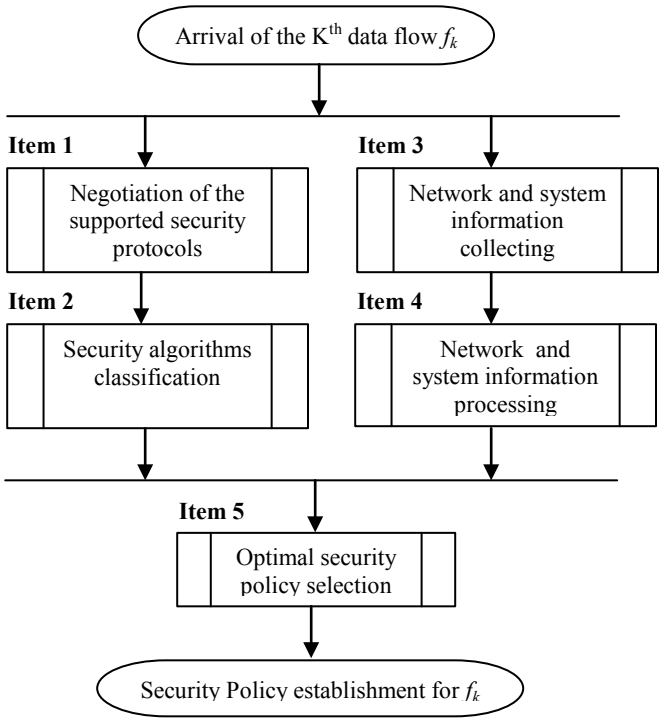

Figure 3. SecMan Flow Chart

In the following subsections, every step in the flow chart is detailed.

Item 1 - Negotiation of the Supported Security Protocols

In *SecMan*, in order to use the AHP algorithm and evaluate the security level of each policy, it is important to establish a set of security mechanisms supported at the same time by the on-board SMP and the ground entity involved in the exchange. As we noted before, these alternatives are beforehand negotiated using a secure protocol defined to establish a common SSPD. The reason why we need such a protocol is that a ground entity may provide some ciphers or protocols which are not supported by SMP, and then the establishment of a secure connection would be impossible. Similar mechanism can be found in other protocols (e.g. Internet Security Association and Key Management Protocol – ISAKMP [21]) but the negotiation phase is

systematically followed by the connection establishment: if we use these protocols, SecMan would be unable to process the decision making algorithm in real time. ISAKMP negotiation phase is also part of the IPSec [22] global framework; using it out of the protocol remains a difficult task.

Moreover, securing this negotiation phase is fundamental: as long as this negotiation is the first step of all the *SecMan* process, it has to be secure to be sure that the global security architecture is not compromised. Previously, we made a vulnerability analysis of the protocol; we found that several attacks such as Man In the Middle (MITM), replay attacks, and masquerading are possible. However, this study is out-of-scope of the paper, and we present only the secure version of the negotiation protocol using an asymmetric cryptographic system based on a Public Key Infrastructure (PKI). In the following description of the protocol, we assume that the PKI exists to check all the credentials (e.g. certificates or public/private keys).

Figure 4 shows the secure negotiation protocol of the supported security mechanisms. Three entities are involved in the negotiation procedure: E_1 is the onboard entity (e.g. a passenger terminal), E_2 is the ground entity (e.g. a server), and the corresponding SMP (c.f. system architecture). E_1 requests a secure connection with E_2, SMP initiates the negotiation in order to establish the SSPD by sending a *Request_SSP* message containing a random number $Nonce_1$ to avoid replay attacks. The SSPD structure storage is the following:

$$\langle IP_d, Port_d, Proto., SSP_{negotiated}, h_{E_2}, Expir.date \rangle$$

where IP_d is the IP address of E_2, $Port_d$ is the destination port, $Proto.$ is the used protocol (e.g. TCP or UDP), $SSP_{negotiated}$ is the common security protocols supported by both onboard and ground entities, h_{E_2} is the hash of SSP_{E_2}, and $Expir.date$ is the expected date when the negotiated security protocols are no longer effective.

At the reception of the request, E_2 computes a hash h_{E_2} of the supported security mechanisms SSP_{E_2} and answers with a *Response_SSP* message containing its own certificate $Cert_{E_2}$, $Nonce_1$, a new generated random number $Nonce_2$, the list of supported security mechanisms SSP_{E_2}, the hash h_{E_2}, a *lifetime* and the signature of the message Sig_{E_2} using the private key of E_2 (to ensure the integrity of the response).

Lifetime allows SMP to define the expiration date of the supported security mechanisms by E_2. Thanks to *lifetime* and hash h_{E_2}, SMP can avoid a negotiation procedure if the supported security mechanisms have not been modified since the last negotiation. The certificate is used by SMP to check the validity of the public key and the authenticity of E_2 with the Certificate Authority (CA). When SMP receives the response, it verifies the certificate, $Nonce_1$ and the signature Sig_{E_2} validities. Then SMP establishes the negotiated security parameters $SSP_{negotiated} = SSP_{SMP} \cap SSP_{E_2}$ and computes the expiration date by adding the reception date to the received *lifetime*. Finally, SMP updates the SSPD (using an *update* message) and establishes a secure connection with E_2, now *SecMan* is able to compute the good security policy.

If a previous negotiation phase has been done with the same entity E_2 and the *lifetime* is no longer valid, SMP requests that E_2 computes another time the hash h'_{E_2} in order to check if the ground entity has modified its supported security mechanisms set: SMP sends a *Request_hash* message with the $Nonce_2$ (to allow E_2 verifying the freshness of the request) and a new generated $Nonce_3$. E_2 answers with a *Response_hash* message containing $Nonce_3$, $Nonce_4$, $Cert_{E_2}$, the new hash h'_{E_2} and the signature Sig_{E_2}. At the receipt of the answer, SMP verifies the validities of all the materials included in *Response_hash* and then continues the negotiation procedure if the hashes are different ($h_{E_2} \neq h'_{E_2}$), else a secure connection is directly established (there is no need to repeat the negotiation procedure) and then additional network resources are not consumed as we are going to see in the following paragraphs.

As noted before, the *lifetime* and hash are used to avoid excessive resources wastage. Actually, the *lifetime* metric is sufficient to know if a new negotiation procedure is needed or not. But, quantitatively, the *Request_hash* and *Response_hash* messages are less bulky than *Request_SSP* and *Response_SSP* messages: a hash size is about few bytes (for instance, SHA-1 hash size is 20 bytes) whereas the SSPD is planned to be stored in XML stream (for instance, an XML stream containing SSH, IPsec, and TLS mechanisms has a 1000 bytes size).

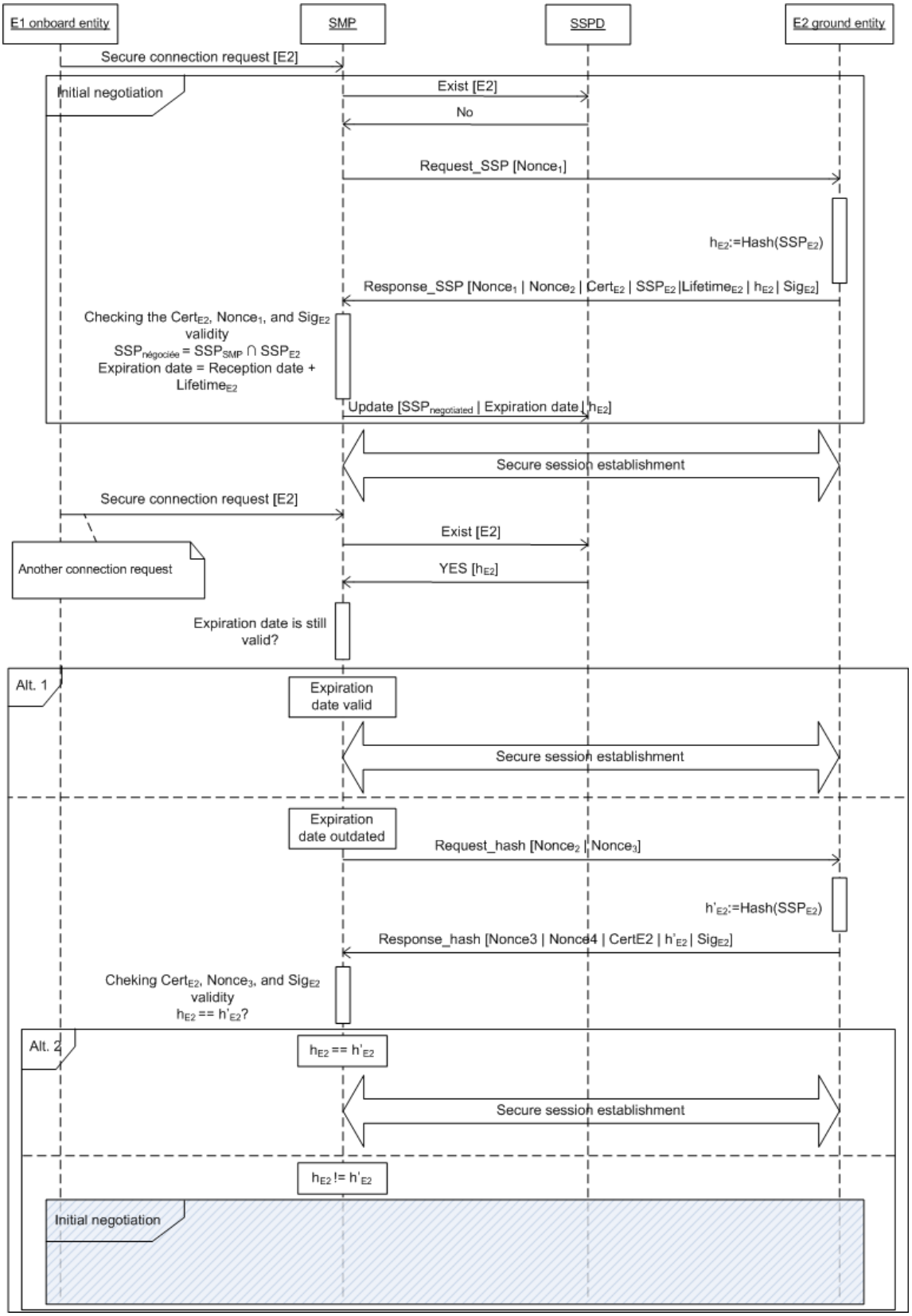

Figure 4. Secure Negotiation of the Supported Security Protocols (Item1)

When the negotiation phase is finished, the $SSP_{negotiated}$ is made available for the AHP block for a classification by security level, network cost and system cost (see the next section).

Item 2 - Security Algorithms Classification: an AHP Approach

AHP [20] is an analytical multi-criteria technique used to resolve decision making problems for complex and heterogeneous systems. Basically, the idea is to find out what are the factors involved and then constitute a hierarchy model according to the association of the factors. The advantages of AHP method are to combine qualitative and quantitative analysis, and then reduce the subjectivity in order to make the evaluation more reliable and to design an automatic process. Previously, AHP has been used in many fields to select alternatives from a set of candidates (social problem, economy or management for instance). Based on its advantages, we decided to adopt AHP in security level assessment for the *SecMan* security classification process.

The method is simple to use and the results are usually relevant. The AHP process is mainly succession of the following steps:

- Establishment of the hierarchical structure,
- Pair-wise comparison (weight criteria and alternatives),
- Consistency evaluation,
- Weights synthesis and score computation of each alternative.

The first step is to identify the main objective behind the use of AHP, criteria, sub-criteria, and set of alternative choices used for the comparison. The alternatives are actually leafs of the tree and the objective corresponds to the root. For instance, Figure 5 shows the hierarchy we used for ranking the security algorithms. Table 1 shows criteria we used in *SecMan*, but the list can be extended with other factors. Some metric values such as the delay for the network cost or the cryptographic throughput are specific to the system cost and the testbed we used when we measured these parameters (cf. last section).

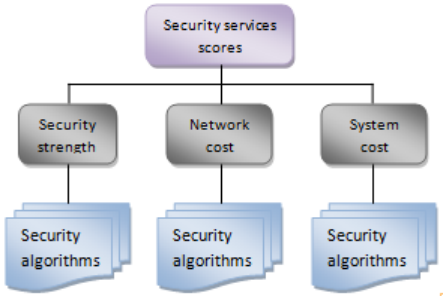

Figure 5. Multi-Criteria Hierarchy Used to Evaluate the Security Algorithms with AHP

Table 1. Criteria and Sub-Criteria List Used in AHP for the *Secman* Framework

Security service	Security strength	Network cost	System cost
Confidentiality	Key length (bytes) Block size (bytes) # rounds (#)	Delay (cryptographic time, signalization overhead) (msec) Packet overhead (bytes)	Cryptographic throughput (MB/s)
Integrity	Hash length (bytes) Block size (bytes) # rounds (#)	Delay (Hash generation time, signature verification, signalization overhead) (msec) Packet overhead (bytes)	
Authentication	Key length (bytes)	Delay (public/private key generation time, signature verification, signature generation) (msec) Packet overhead (bytes)	

The second step is the pair-wise comparison. Pair-wise comparisons are performed at each level of the hierarchy, starting from the bottom of the tree, with respect to the upper level objectives. Once all the alternatives of the same level are compared, the evaluation is moved to the upper level.

A specific scale is used to perform such comparisons (ranking from 1 to 9), and called AHP ratio scale: value 1 corresponds to an "equally important" scale and value 9 corresponds to an "extremely important" scale. The other values within the AHP ratio scale have intermediate meaning between these two bounds.

A square comparison matrix $D = (d_{ij})$ of order n is then established with the constraints that $d_{ij} = 1/d_{ji}\ \forall\ i \neq j$, and $d_{ii} = 1, \forall\ i$(the matrix D is said to be reciprocal). The weights d_{ij} are consistent if they are transitive, that is $d_{ik} = d_{ij} \times d_{jk}$ $\forall\ i, j$ and k. The consistency is the logical coherence among the judgments (and in this case between the weights): for example, if an object *A* has greater value than an object B (we write *A* > *B*), and *B* has greater value than an object *C* (*B* > *C*), then logically *A* has greater value than *C* (*A* > *C*). This logic of preference is called transitive property. Thus consistency is closely related to the transitive property of the matrix D.

The next step is to find a vector ω of order n such that $D\omega = \lambda\omega$. For such a matrix, ω is said to be an eigenvector and λ is an eigenvalue. For a consistent matrix, λ is equal to n . For matrices involving human judgment, the condition $d_{ik} = d_{ij} \times d_{jk}$ does not hold as human judgments are more or less inconsistent. In such a case the ω vector satisfies the equation $\omega = \lambda_{max}\omega$ and $\lambda_{max} \geq n$. The difference, if any, between λ_{max} and n is an indication of the inconsistency of the judgments. If $\lambda_{max} = n$ then the judgments have turned out to be consistent. Finally, a Consistency Index (*CI*) is calculated: $CI = (\lambda_{max} - n)/(n - 1)$. That needs to be assessed against judgments made completely at random and a true Consistency Ratio (*CR*) is calculated by dividing *CI* for the set of judgments by the Index *CI* for the corresponding random matrix.

If *CR* ≥ *0.1,* the judgment may be too inconsistent to be reliable and the exercise must be repeated. A *CR=0* means that the judgments are perfectly consistent, but practically a *CR* ≤ *0.1* is sufficient. These are the fundamental mathematics used in AHP. For more details and few applications of AHP, [20] is recommended.

In *SecMan* context, AHP is applied to the negotiated security algorithm a_{ij}^{s} set of every security mechanism m_j . a_{ij}^{s} is the i[th] security algorithm offering the security service *s* and being used by the security mechanism m_j. For instance, if the security mechanism m_j is SSL (Socket Secure Layer) [23], DES(128)-CBC (Data Encryption Standard using a 128 bit key and the Cipher Block Chaining mode) [24] is a possible security algorithm a_{ij}^{conf} .

The outputs of this step are a security level (strength) score per service *s*, a network cost and system cost scores for every a_{ij}^{s} / $s \in S = \{confidentiality, integrity, authentication\}$.

The algorithm applied in item 2 is:

Item2
For every security mechanism m_j **do**
For every security service *s* **do**
Apply AHP for the set $\{a_{ij}^{s}\}$ to establish the security level score $V_{sec}^{s}(a_{ij}^{s})$, the network cost score $V_{cost}^{net}(a_{ij}^{s})$ and the system cost score $V_{cost}^{syst}(a_{ij}^{s})$
End For
End For

Item 3 - Network and System Information Collecting

In order to get network or system information and make the optimal choice, the Simple Network Management Protocol (SNMP) [25] is used: network and system metric values are sent deploying SNMP local agents on each monitoring entity of the system. For instance on each onboard router, information such as *ifSpeed* for the available bandwidth are retrieved from the Management Information Bases (MIBs) by SNMP agents (A_{NG} or A_{ATN} for instance) and sent to a Network Management System (NMS) located on the SMP (see figure 1 for details about SNMP agent deployment and location in the communication architecture).

For optimization purposes, information are sent only when a new security policy for a new coming data stream has to be established. We have eliminated SNMP polling-based collecting technique in order to reduce excessive usage of network resources. Then, the information exchange is initiated when a new data flow has to be secure. The extra delay for the user application induced by this transfer remains low given that network or system information are only exchanged on the aircraft segment of the whole communication. It means that only entity such as ATN or NG routers, SMP or satellite terminal can be involved in this information transfer. Indeed, we consider that both network and system bottlenecks are onboard or on the air-ground satellite link and so we do not need to exchange network or system information with ground entities. Note that information on the air-ground satellite link is provided by the onboard satellite terminal.

Also a "*lifetime_info*" metric is defined in case many data streams follow one another in short time. Another policy may be to keep an eye on the network state but this is useless as long as *SecMan* needs these information only when a security policy establishment is required. The "*lifetime_info*" value is a function of data stream arrival rate. In fact, choosing a fixed value can be a problem: a small value can be the cause of an excessive number of SNMP requests. If a big value is chosen, the network state may change while *SecMan* would not be necessarily informed by these variations. On the other hand, the use of the third version of SNMP is recommended: requests and responses are secure preventing intrusions when network or system information are exchanged. The algorithm designed for item 3 is described below:

Item3

If k =1 **then** *//first data stream*

$lifetime\_info \coloneqq 1$ *//1 second*

Else

$lifetime\_info \coloneqq \frac{1}{2} lifetime\_info + \frac{1}{2}(t_k - t_{k-1})$ *// t_k and t_{k-1} are respectively the k^{th} and (k-1)th f_k and f_{k-1} arrival date*

$lifetime\_info \coloneqq Max(lifetime\_info, 45ms)$ *// this is the minimum boundary, 45 ms is the time to the update information frequency used in the DAMA [1] protocol* [16]

$lifetime\_info \coloneqq Min(lifetime\_info, 10\,\text{sec})$ *// this is the maximum boundary used when many simultaneous data stream have to be managed*

End if

If $lifetime\_info$ is no longer valid **then**

Send SNMP request *//get information network from the MIBs with the corresponding Object IDentifier (OID)*

Update $lifetime\_info$

Else

Keep using the same network information retrieved for the (k-1)th data stream.

End if

In the next step, the collected information are used to define the network and system constraints which need to be respected.

Item 4 - Network and System Information Processing

Here, the information received in **Item2** are used to define the network and system constraints before the security policy selection phase. Let $\Omega_{c,net}^{used} \in [0,1]$ the used network resources ratio by the data stream of the class c / c ∈ C={APC, AOC, AOC NG, ATS}. The priorities between the services classes are:

$$\text{ATS} > \text{AOC NG} > \text{AOC} > \text{APC}$$

$\Omega_{c,net}^{available} \in [0,1]$ is the available network resources ratio for the data stream of the class c, this metric is defined thanks to the information of the class c and the classes with lower priorities (for instance, if c is AOC NG, then the information relative to AOC and APC are taken into account to define $\Omega_{c,net}^{available}$). The same metrics are defined for the system resources (namely $\Omega_{c,syst}^{used}$ and $\Omega_{c,syst}^{available}$).

For example, if $\Omega_{c,syst}^{available} = 0$, then the network resources are no longer available for the applications belonging to class c. In this paper, we consider only the bandwidth as a network resource and the CPU as a system resource. The following equations allow us to define available resources:

$$\Omega_{c,net}^{available} = 1 - \sum_{i \in C, i \geq c} \Omega_{i,net}^{used}$$

$$\Omega_{c,syst}^{available} = 1 - \sum_{i \in C, i \geq c} \Omega_{i,syst}^{used}$$

For example, if: $\begin{cases} \Omega_{ATS,net}^{used} = 0.1\ (10\%) \\ \Omega_{AOC,net}^{used} = 0.2\ (20\%) \\ \Omega_{APC,net}^{used} = 0.3\ (30\%) \end{cases}$ then:

$$\begin{cases} \Omega_{ATS,net}^{available} = 1 - \Omega_{ATS,net}^{used} \\ \qquad = 0.9\ (90\%) \\ \Omega_{AOC,net}^{available} = 1 - \left(\Omega_{ATS,net}^{used} + \Omega_{AOC,net}^{used}\right) \\ \qquad = 0.7\ (70\%) \\ \Omega_{APC,net}^{available} = 1 - \left(\Omega_{ATS,net}^{used} + \Omega_{AOC,net}^{used} + \Omega_{APC,net}^{used}\right) \\ \qquad = 0.4\ (40\%) \end{cases}$$

We can clearly see in this example that the priorities between the class services are respected (cf. inter-class and intra-class modes introduced above). In order to calculate the network and system available resources, we need first to define $\Omega_{c,net}^{used}$ and $\Omega_{c,syst}^{used}$ using the informations obtained in **Item2**:

$$\Omega_{c,net}^{used} = \frac{BW_c^{used}}{BW_c^{used} + BW_c^{available}}$$

where BW_c^{used} and $BW_c^{available}$ are respectively the used and available bandwidth ratio for the

[1] Demand Assigned Multiple Access (DAMA) protocol is used to improve the usage efficiency of the available transmission resource (TR) and allow multiple users to share the same link bandwidth on the satellite link for FAST project.

applications of class c. For $\Omega_{c,syst}^{used}$, which is system specific (and a characteristic of the SMP where *SecMan* is intended to run), we use the data stream per class in order to find out the CPU local occupation per class:

$$\Omega_{c,net}^{used} = \%CPU \times \frac{N_c}{N_{total}}$$

where N_c is the number of data stream of class c and N_{total} is the total number of considered data stream. Recall that input parameters such as BW_c^{used}, $BW_c^{available}$ or %CPU are based on environment information collecting function provided by item 3. The algorithm designed for item 4 is:

Item4

For every service class $i \geq c$ **do**

$$\Omega_{i,net}^{used} = \frac{BW_i^{used}}{BW_i^{used} + BW_i^{available}}$$

$$\Omega_{i,syst}^{used} = \%CPU \times \frac{N_i}{N_{total}}$$

End For

$$\Omega_{c,net}^{available} = 1 - \sum_{i \in C, i \geq c} \Omega_{i,net}^{used}$$

$$\Omega_{c,syst}^{available} = 1 - \sum_{i \in C, i \geq c} \Omega_{i,syst}^{used}$$

Item 5 - Optimal Security Policy Selection

The aim of SecMan at this final phase, is to establish a security policy P_k for the data stream f_k. Formally, P_k can be represented as:

$$P_k = [p_{k,0}, p_{k,1}, \ldots, p_{k,n}]$$

where $p_{k,j} \in \{0,1\}$ and $n = card(SSP_{negotiated})$: if $p_{k,j} = 1$, then the security mechanism m_j is selected in P_k. Figure 6 is a pyramidal representation of the relationships between a_{ij}^s, m_j, and P_k.

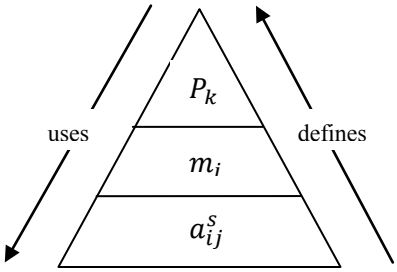

Figure 6. Pyramidal Security Concepts Representation

Let $B_k = [b_k^{conf}, b_k^{int}, b_k^{auth}]$ the minimum security requirements vector expressed for the data stream f_k / b_k^s is the elementary security requirement for the security service *s*. The security requirements have been assessed using the risk analysis in [4] for the ATS and AOC data-based services. The APC and AOC NG services security needs have been assessed according to the main users. Table 2 shows the security requirements used per class:

Table 2. On Board Service Security Requirements

Class	b_k^{conf}	b_k^{int}	b_k^{auth}
ATS	2	4	4
AOC	2	3	2
AOC NG	2	3	2
APC	2	2	1

Table 3 the elementary security requirements ratio scale. The security requirements are now mapped with the security strength scores calculated in **Item2** then the optimal security policy is extracted. The optimal security policy has to respect the following constraints:

- The security requirement vector B_k must at least be satisfied;
- The network and system constraints computed at **Item4** must be verified.

Table 3. Security Requirements Description

Security requirement value	Description
0	No security need
1	Low
2	Medium
3	High
4	Very high

The optimal security policy for the data stream f_k is denoted P_k^* and corresponds to the security policy which maximizes the security level while the network and system costs are minimized:

$$P_k^* = argmax_{P_k}\{SPL(P_k) - \beta \times Cost(P_k)\}$$

Where the following constraints are verified:

- B_k is verified by P_k^* ;
- $Cost_{net}(P_k^*) \leq \Omega_{c,net}^{available}$;
- $Cost_{syst}(P_k^*) \leq \Omega_{c,syst}^{available}$.

$\beta \in \mathbb{N}^*$ is a positive coefficient to balance the security policy level and the cost of P_k. *SPL* is the Security Policy Level of P_k mapped with the security requirements of f_k and $Cost(P_k)$ is the overall cost value of P_k :

$$PSL(P_k) = B_k^T \times V_{sec}(P_k);$$

$$Cost(P_k) = Cost_{net}(P_k) \times \left(1 - \Omega_{c,net}^{available}\right) + Cost_{syst}(P_k) \times (1 - \Omega_{c,syst}^{available}).$$

The cost and security strength values are established using the AHP computed scores:

- $Cost_{net}(P_k) = \sum_{j=0}^{n} p_{k,j} \times V_{cost}^{net}(m_j)$;
- $Cost_{syst}(P_k) = \sum_{j=0}^{n} p_{k,j} \times V_{cost}^{syst}(m_j)$;
- $V_{sec}(P_k) = \sum_{j=0}^{n} p_{k,j} \times V_{sec}(m_j)$.

The $V_{cost}^{net}(m_j)$, $V_{cost}^{syst}(m_j)$ and $V_{sec}(m_j)$ values are deduced from the AHP scores of the security algorithms a_{ij}^{s} (outputs of **Item2**). Let Relative Balance Index (RBI) of a security policy: $RBI(P_k) = SPL(P_k) - \beta \times Cost(P_k)$.

Finally, the algorithm for optimal security policy section is introduced below:

Item5

$P_k^* = [0, 0, \ldots, 0]$; *//initialize the optimal security policy to null*

$RBI(P_k^*) = -\infty$;

For every P_k **do**

If B_k satisfied by P_k **then**

Compute *PSL* (P_k);

Compute $Cost(P_k)$;

$RBI(P_k) := PSL(P_k) - \beta \times Cost(P_k)$;

If $Cost_{net}(P_k) \leq \Omega_{c,net}^{available}$ and $Cost_{syst}(P_k) \leq \Omega_{c,syst}^{available}$ and $RBI(P_k^*) \leq RBI(P_k)$ **then**

$P_k^* := P_k$;

$RBI(P_k^*) = RBI(P_k)$;

End If

End If

End for

Testbed Infrastructure

We designed a testbed platform to implement and validate the different SecMan mechanisms detailed previously. The main goal of this environment is to emphasize advantages of adaptive selection for security mechanisms assignment.

Our testbed platform uses Marionnet environment which aims at emulating our different systems (ATS/AOC/APC clients, ATS/AOC/APC servers, SecMan Proxies, ATN/IPS and NG routers and satellite connection). Marionnet environment implements Linux User Mode to run the different virtual machines, more information on this development method can be found in [26].

Evaluation Scenarios

We defined two different scenarios. The first one validates intra-class operational mode for SecMan by focusing on a specific traffic class of FAST project: Air Traffic Services (ATS). The second one validates inter-class operational mode by putting in concurrence two different traffic classes: AOC and APC.

In our experiments, we consider different application flows; each one is generated with WGET application (http://www.gnu.org/software/wget/) with a reference throughput equal to 800 kbps. These experiments do not attempt to represent in an accurate way ATS, AOC or APC traffic that is why every flow has the same throughput.

Our main goal is to validate improvements for resource allocation based on our adaptive security mechanism management. To do so, we increase the number of ATS, AOC or APC flows exchanged between ATS, AOC or APC client and server through the ATN/IPS or NG router and each SMP. For each new flow, SecMan selects the best security mechanism to deploy in order to satisfy initial ATS, AOC or APC application security needs and also system and network resources available when the connection needs to be established.

Experimental Results and Discussion

Scenario 1: SecMan Intra-Class Operational Mode

Figure 7 represents for each new ATS flow, the robustness level provided by SecMan SMP_2 according to ATS application security needs and the different security mechanisms it can select based on its SSPD. In this scenario, to satisfy application security needs and based on network and system resource level, SecMan's decision is to select between 4 different security policies.

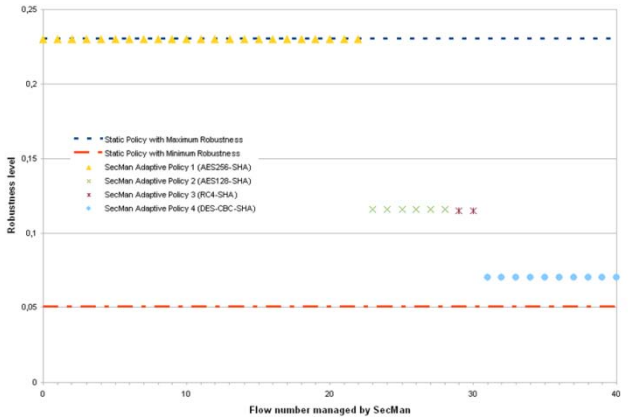

Figure 7. Robustness Level for the Different ATS Flows Managed by SecMan

Note that SSPD has many more available security mechanisms but in this scenario SecMan do not need to use all of them to optimize resource and security level. Moreover, there are two references in this chart; first one is the maximum robustness which one flow can get through SecMan according to the most secure mechanism available in the SSPD.

The second one is the minimum robustness which SecMan can provide according to the less secure mechanism it can pick up in the SSPD. We can see in figure 7 that SecMan adapts for each new ATS flow the robustness level it provides thanks to its adaptive algorithm. The robustness is always better than the minimum robustness level considering initial application security needs but not equal to the maximum level of robustness given than this last value is not the optimal one for network and system resource allocation (as we are going to see in the next paragraph).

Indeed, Figure 8 illustrates improvements for network and system resource allocations. We compare on the same chart network (with network security mechanism overhead evolution on the upper part of the figure) and system (with consumed CPU percentage on the lower part) resource allocations provided by the adaptive SecMan management and two static security policies based on the minimum and the maximum level of robustness that SSPD can provide.

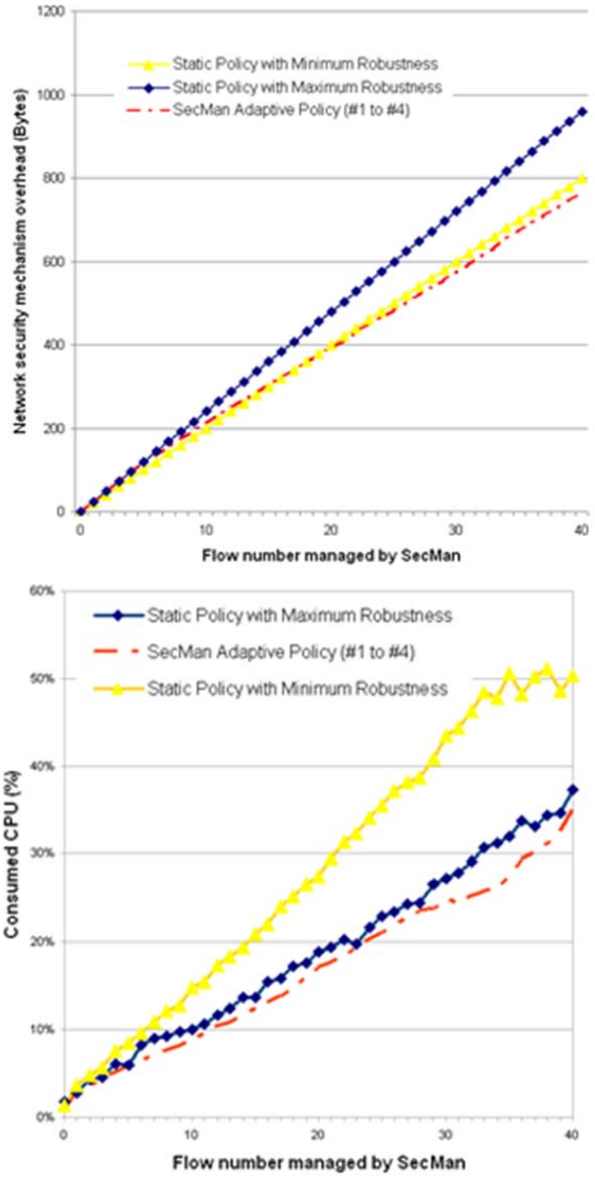

Figure 8. Network and System Resource Allocation Improvements with SecMan Management

We can easily notice that SecMan uses less resources (both at network and system side) than a static security policy. Then, SecMan allows serving more ATS flow than a traditional static security policy assignment. Even if we consider the less robust policy, resource consumption is still more important than with SecMan adaptive policy. This last result is important because it shows that robustness level and consumed resources do not vary on the same way and then, it is not possible to select, a priori, a static policy to optimize resource allocation.

In this scenario, global improvement with adaptive security management is on average respectively 15% for network resources and 7% for system resources, so more ATS application flows can be serve on the aircraft with the same available network capacity and CPU than with traditional and static security mechanism assignment.

Scenario 2: SecMan Inter-Class Operational Mode

In this second configuration, we put in concurrence different flow classes (AOC and APC applications). This is a realistic scenario for SMP_1 behavior which is connected to NG router and manages such traffic.

Figure 9 depicts network resource allocation for the different traffic classes. We can see that priority is managed between the different classes. AOC always get resources to be exchanged but APC may be under provisioned according to the QoS policy previously introduced. Note that for this scenario, SMP_1 configuration allows a maximum provisioning for AOC traffic of 60%, that is why maximum network capacity for AOC traffic is not above this value. Also, in such configuration the SecMan is still able to maximize the robustness level (see figure 10 for details) according to available resources and traffic class priorities.

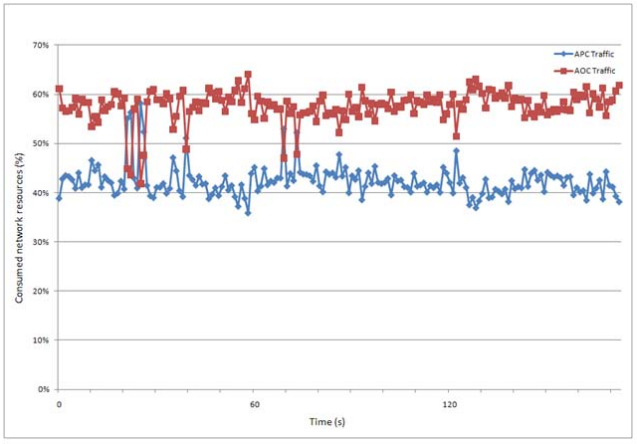

Figure 9. Network Resource Allocation with SecMan Management in Inter-Class Operational Mode

Figure 10. Robustness Level for Flows Managed by SecMan in Inter-Class Operational Mode

Conclusion

In this paper, we have introduced a novel adaptive security management approach for future connected aircrafts. The security management algorithm is formalized and the global architecture performances are evaluated. Some points have to be resolved although.

First, scaling issues were not considered in the paper. The international air traffic and the increasing carried passenger's number are factors which need to be taken into account: on-board entities may need to exchange or check the validity of used certificates or keys. This requires the use of a PKI adapted to the aeronautical specific context. This PKI is also fundamental to secure the negotiation protocol and build the SSPD introduced in this paper.

Moreover, scalable available resource monitoring techniques have to be investigated. Indeed, in this paper we consider only one aircraft deploying SecMan but with several aircrafts "SMP compliant" we can imagine improvements for our QoS based adaptive security management policy.

Finally, using the security architecture in a real airborne environment cannot be done without a certification procedure: software considerations in airborne systems and equipment certification are described in the Radio Technical Commission for Aeronautics (RTCA) and EUROCONTROL DO-178B document [27]. Thus, we are considering the use of a high-assurance design technique called

MILS (Multiple Independent Layer Security and Safety) [28] to implement the *SecMan* module.

References

[1] EUROCONTROL, May 2004, Network Operations Report 2009 – Indicators and Analysis of the ATM Network Operations Performance.

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

[3] Shingledecker, C. and Giles, S. and Darby, E.R., Jr. and Pino, J. and Hancock, T.R., 2005, Projecting The Effect of CPDLC on NAS Capacity, Digital Avionics Systems Conference, 2005. DASC 2005. The 24th, 8 pp. Vol. 1.

[4] EUROCONTROL, FAA, 2007, Communications Operating concept and Requirements for the Future Radio System (COCR), Version 2.0.

[5] Radzik, .J and Pirovano, A. and TAO, N. and Bousquet, M., 2007, Satellite system performance assessment for In-Flight Entertainment and Air Traffic Control, Space Communications journal, special issue on Satellite Networks for Mobile Services.

[6] ICAO, 2002, Manual for the ATN using IPS Standards and Protocols, Document 9896.

[7] Getachew, D., Phd and Griner, J. H., 2005, An Elliptic Curve Based Authentication Protocol For Controller-Pilot Data Link Communications, International Journal of Computer Science and Network Security.

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

[9] Olive, M.L., 2001, Efficient Data Link security in a Bandwidth-limited Mobile Environment An Overview of the Aeronautical Telecommunications Network (ATN) Security Concept, Digital Avionics Systems Conference, 2001. DASC. The 20th Conference, 9E2/1-9E2/10 vol.2.

[10] Cruickshank, H. and Howarth, M.P. and Iyengar, S. and Zhili Sun and Claverotte, L., 2005, Securing multicast in DVB-RCS satellite systems, Wireless Communications, IEEE, vol. 12, 38 – 45.

[11] Iyengar, S. and Cruickshank, H. and Pillai, P. and Fairhurst, G. and Duquerroy, L., 2007, Security requirements for IP over satellite DVB networks, Mobile and Wireless Communications Summit, 2007. 16th IST, 1 -6.

[12] Duquerroy L., Josset S., Alphand O., Berthou P. and Gayraud T., 2004, SatIPSec: an optimized solution for securing multicast and unicast satellite transmissions", 22nd AIAA International Communications Satellite Systems Conference, Monterey.

[13] Ben Mahmoud, MS. and Larrieu, N. and Pirovano, A, 2009, An aeronautical data link security overview, Digital Avionics Systems Conference, 2009. DASC '09. IEEE/AIAA 28th, 4.A.4-1 -4.A.4-14.

[14] ETSI, 2005, Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broaband satellite applications, EN 302 307, V1.1.1.

[15] ETSI, 2005, Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems, EN 301 790, V1.4.1.

[16] Ben Mahmoud, MS. and Larrieu, N. and Pirovano, A., 2010, Security Architecture Design for Satellite Aeronautical Data Link communications, 28th AIAA International Communications Satellite Systems Conference . ICSSC 2010.

[17] ETSI, 2006, Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia Services and Architectures: QoS Functional Architecture, Draft TS 102 462, V0.4.2.

[18] ICAO, 2002, Manual of technical provisions of the ATN, Document 9705, Edition 3.

[19] Figueira J. and Greco S. and Ehrgott M., 2005, Multiple Criteria Decision Analysis: State of the Art Surveys, Springer Verlag, Boston, Dordrecht, London.

[20] Saaty, Thomas.L., 2000, Fundamentals of the Analytic Hierarchy Process, RWS Publications, 4922 Ellsworth Avenue, Pittsburgh, PA 15413.

[21] Maughan D., Schertler M., Schneider M., Turner J., 1998, Internet Security Association and Key Management Protocol (ISAKMP), RFC 2408, IETF.

[22] Kent S. and Seo K., 2005, Security Architecture for the Internet Protocol (IPsec), RFC 4301, IETF.

[23] Dierks T., Rescorla E., 2008, The Transport Layer Security (TLS) Protocol, Version 1.2, RFC 5246 (Proposed Standard), IETF.

[24] National Institute of Standards and Technology, Federal Information Processing Standards Publication, 1999, Data Encryption Standard (DES).

[25] Harrington, D. and Wijnen, B., 2002, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks, RFC 3411 (Proposed Standard), IETF.

[26] Loddo, J.V., Saiu, L., 2008, Marionnet: a virtual network laboratory and simulation tool, SimulationWorks, Marseille (France).

[27] RTCA,1992, Software Considerations in Airborne Systems and Equipment Certification DO178-B document.

[28] Jacob, J.M., 2004, High assurance security and safety for digital avionics, Digital Avionics Systems Conference, 2004. DASC 04. The 23rd, 8.E.4-8.1-9 Vol.2.

Email Addresses

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

Nicolas Larrieu: Nicolas.Larrieu@enac.fr

Alain Pirovano: Alain.Pirovano@enac.fr

Antoine Varet: Avaret@recherche.enac.fr