Design and development of an embedded aeronautical router with security capabilities

Antoine Varet, Nicolas Larrieu

DESIGN AND DEVELOPMENT OF AN EMBEDDED AERONAUTICAL ROUTER WITH SECURITY CAPABILITIES

Antoine Varet, Nicolas Larrieu, TELECOM/ResCo research team, Ecole Nationale de l'Aviation Civile (ENAC), Toulouse, France

Keywords = router, embedded, network security, MILS, avionic safety

Abstract

Aeronautical communications are growing in complexity: avionic systems are more and more connected together through complex networks inside and outside the aircraft. This increase of connections requires highly secure systems to solve the constraints of such network topologies. In this paper, we present the IP based Secure Next Generation Router (SNG router) we have developed, providing regulation, routing, secure merging of different data sources and preserving of their segregation. During the design process we searched for a set of solutions to minimize the certification, design and development costs and maintain a high level of security. This paper is organized as follow: we first introduce the methodology we applied to the SNG router and its benefits. Then we examine the partition in charge of the security of data exchanged through the router. The paper is finished with the validation of our router's implementation and with the evaluation of our router's performances.

Introduction

New aeronautical traffic profiles such as the In-Flight Entertainment (IFE), Aircraft Operational Communications (AOC) and Aeronautical Traffic Communications (ATC) are growing in usage and complexity. Higher throughputs and new opportunities could be served by multiplexing this different data but the heterogeneity of their security needs such as confidentiality, integrity, authentication or availability and the need for independence and non-interference remain the main problem for promoting multiplexing solutions through a unique network link.

In order to overcome these issues, we are producing an IP based Secure Next Generation Router (SNG router) in collaboration with the firm Thales Avionics France, providing regulation, routing, secure merging of different data sources and preserving their segregation. During the design process we searched for a set of solutions to minimize the certification, design and development costs and maintain a high level of security.

Thus, this paper is a progress of our previous research work [1] with the presentation of the new methodology we used for the process of aeronautical software development: the goal of the methodology is to rapidly transform verifiable models into a safe and secure byte-code certifiable at DO-178B assurance level A (DAL-A) with reduced costs. The methodology tool chain uses a qualified model transformer to generate code for a secure virtualization infrastructure with controlled inter-partition communications compatible with the ARINC653-APEX standard. A separation kernel running on an embedded hardware target enforces the segregation of computations done on the data.

This paper introduces the application of the methodology to the SNG router, and then details the partition in charge of the data security exchanged through the router. This partition has been designed through the Internet Protocol Security (IPsec [2]) framework and its associated protocols to establish and use secure communication channels. Implementation is based on two approaches: one is the classical application of the methodology while the second consists of linking the design with source code imported from other software projects. The paper is completed with the validation of this implementation and a synthesis of both approaches benefits.

Certification in a nutshell

The aeronautical world requires high levels of assurance in safety for all its systems. The DO-178B [3] standard is the most commonly used document to certify avionic-embeddable software and qualify development and verification tools. It defines 5 design assurance levels, from the lowest and

unconstrained level DAL-E up to the highest level DAL-A which requires exhaustive tests and intensive verifications with independence between the developers and the certifiers.

A MILS-based security approach

The SNG router design is based on a strong functional decomposition into partitions enforced by a Multiple Independent Levels of Security (MILS) compliant operating system. The MILS principles ensure the secure management of data at different levels of security on the same product with a high degree of assurance. Indeed, MILS enforces the N.E.A.T principles: the security solution is Non-bypassable, Evaluatable, Always invoked and Tamperproof. Moreover, the Common Criteria for Information Technology Security Evaluation (CC [4]) standard can be employed to evaluate our product security: the use of models and MILS segregation principles eases security evaluations at high Evaluation Assurance Levels (EAL) defined by the CC. A SNG router Protection Profile has been written for this purpose and is detailed later in this paper.

Different operating systems are MILS compliant. The most common embedded system is probably the WindRiver VxWorks [5] MILS Real-Time Operating System (RTOS). The SNG router we are developing will use a concurrent RTOS, PikeOS, afforded by the Sysgo Company. Sysgo PikeOS [6] is provided with a dedicated Integrated Development Environment (IDE) called CODEO and based on Eclipse. Comparing to the VxWorks IDE named Workbench, CODEO is easier to use than the one provided by WindRiver: we found it more ergonomic because of its tab approach for all configuration items.

Securing the network streams

Aeronautical communication needs security. This generic term “security” covers different complementary aspects; in our case we pointed out three of them. Confidentiality is one of them: information must not be known by non-authorized people. Integrity is another one: data must not be altered between the sender and the receiver; any unattended modification shall be detected. Authenticity of information is the third one: the receiver must be sure that the information comes from a legitimate user and has not been generated by another (malicious) one who has impersonated the right sender.

A Protection Profile has been elaborated during the SNG router specification stage. This document conforms to the Common Criteria standard and contains information on the usage and design of the router. Inside it, additional requirements dedicated to router security follow a vulnerability analysis and the associated security objectives. This Protection Profile is useful to produce a Security Target document for the embedded product. The Security Target will facilitate for both evaluators and industry engineers the security assessment of the final product. Thus Protection Profiles can be considered as templates for generic products, and the Security Targets are the document for the commercialized ones.

Network data can be secure at different levels. Some protocols such as the Secure Shell protocol (SSH [7]) secure the data at the application level, whereas others, such as Wi-Fi Protected Access (WPA & WPA2 [8]), secure the radio communications at the link layer. The Internet Protocol Security, most commonly called IPsec[2], is a framework to secure the data at the network level, above the physical and link layers and under the transport layer as illustrated in figure 1. A router interacts at the network level too, so IPsec is a privileged framework to integrate the security into the SNG router.

This framework is based on different protocols. The Internet Key Exchange protocol (IKE [9]) is a protocol to create and establish a secure channel. The channel can be a tunnel which encapsulates IP packets or a layer to insert additional transport data to packets. Currently, IKE is rendered obsolete by IKEv2 [10], the next, improved, simpler and less vulnerable version of IKE.

Another part of the IPsec framework is Encapsulating Security Payload (ESP [11]). This IPsec mechanism encrypts and encapsulates the data to secure, adding integrity and authentication data. The receiver uses this information to validate the received data. The SNG router data security design is based on the IKEv2 and ESP protocols.

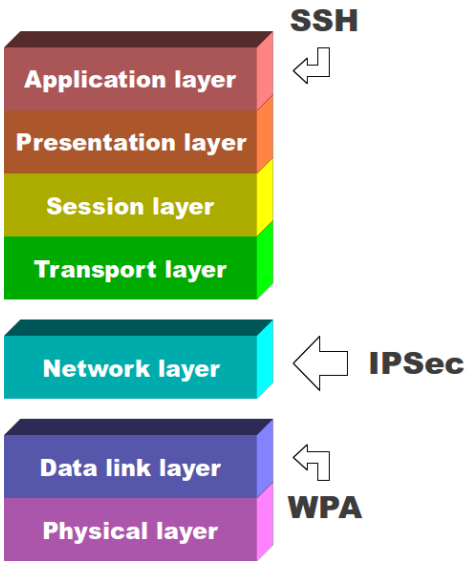

Figure 1 Some security protocols on the OSI model

Development methodology

During the SNG router development process, we elaborated a methodology to accelerate the development with several process optimization goals. First of them is the minimization of costs and delays for design, source code writing, certification and evaluation. The second group of benefits concerns maximization of the safety and security assurance levels in the confidence of the final software.

This methodology summarized in figure 2 is explained in [1]. It is based on seven different steps. During the partitioning first step, the software architect decomposes the specification into different subsets of requirements, each subset being associated to its "partition". Then for each partition the designer designs a high-level model, refining the subset of functionality into a more concrete (graphical) source code. During the third step, each high-level model is converted into library source code, a code independent of any operating system or hardware architecture. The library source code (from the 3rd step) has to be adapted to a specific operating system: the glueing fourth step consists of generating this code. For each partition, the library and glue source codes are compiled together into partition binary files during the compiling fifth step.

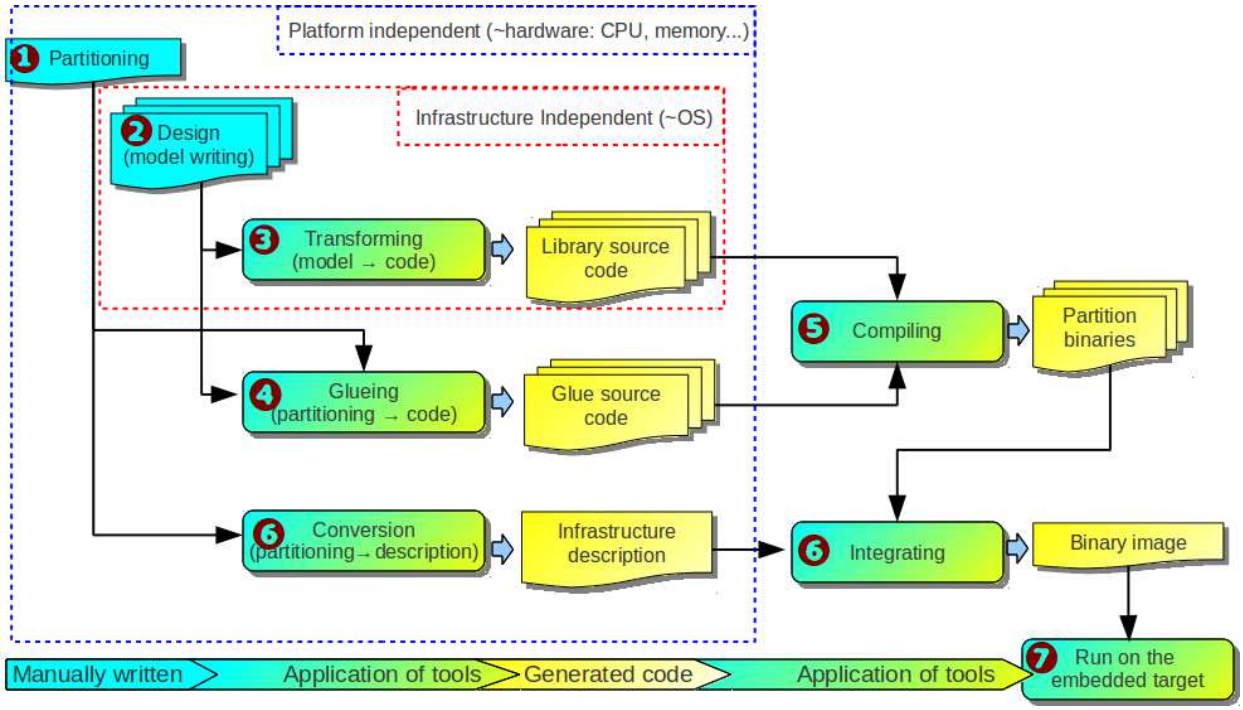

Figure 2 Development methodology used for the SNG router

Then all binary files of all partitions are aggregated with the operating system binary in the sixth integrating step into one final binary image, conforming to the plans defined during the first step by the architect. The last seventh step consists of running the final image or emulating it for verification and validation.

The SNG router: from the specification to the source code

Router inputs, outputs and validation experiments

The SNG router we are developing has several Ethernet network interfaces for routing and securing IP packets. The entire configuration is defined during the router installation inside the aircraft and cannot be modified afterwards. The configuration data blocs contain static IPv4 and IPv6 addresses, static routing tables and static filtering rule tables. These data blocs cannot be modified during the flight. This immutability makes final product certification easier and shorter but the router may be more dynamic in future versions to increase its scalability. Packet security is enforced within aeronautical networks by using two or more SNG routers in concert.

Figure 3 shows the validation topology: a topology with 2 SNG routers, 2 "critical" domains named Airline Information Services Domain (AISD) which require safety and security, 2 "highly critical" ones called Aircraft Control Domain (ACD) which require a higher level of safety and security and a 5th non-critical domain without any requirement. Each router has 3 network interfaces. The routers communicate together with a tunnel through the non-critical domain and route packets between the other domains, preserving their requirements in terms of safety and security.

Any of the 2 SNG routers incorporated in the topology are linked with secure tunnels through an unsecured domain. They enable the highly critical grey domains to communicate securely through their dedicated secure tunnel, i.e. to exchange IP packets with confidence, integrity and authentication checks. This is similar for the critical yellow domains. The separation given by the SNG router is another positive point: if a host of the yellow critical domain is corrupted, the hosts of the grey highly critical domains and their network communications are not infected.

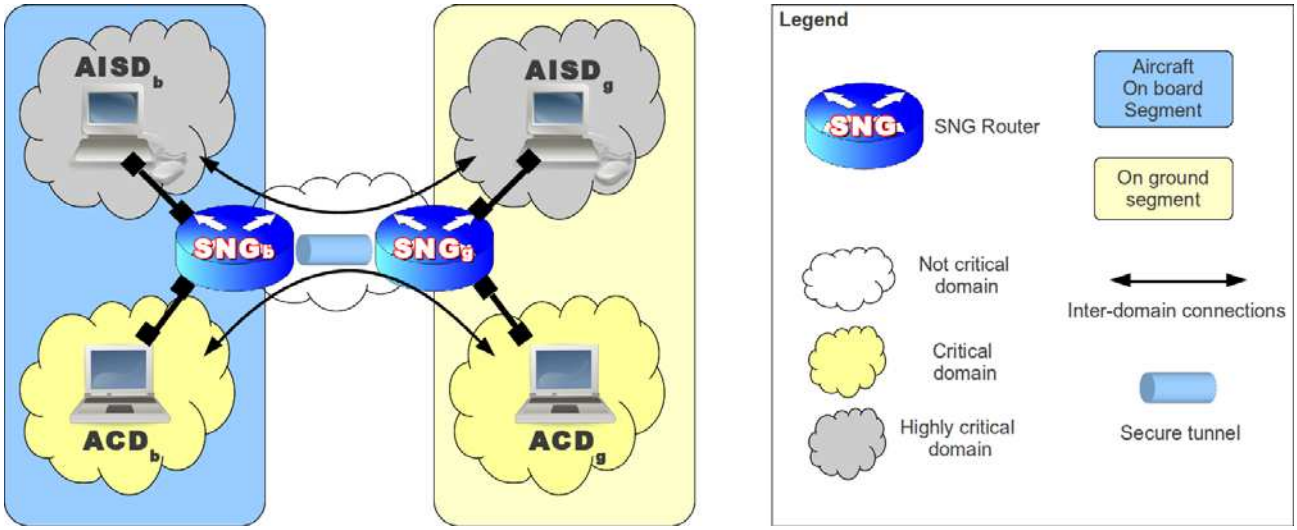

Figure 3 SNG router validation topology

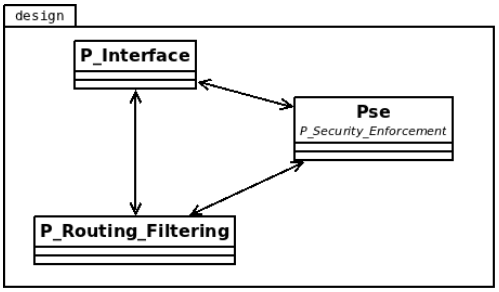

Figure 4 Simplified design of partitions for SNG routers

Methodology application for our router

To build the SNG router software, the architect defines different partition classes, each class being associated to one or more functionalities and implemented by designers.

One class can be instantiated once or multiple times; for example in figure 4 presented above, the P_interface is a partition class to link the hardware with the software parts for any network interface, designed only one time for any number of external network interfaces of the router.

The example configuration presented in figure 5 has 3 instances of the P_interface class: Interface1, Iface2 and Iface3. As another example, figure 5 describes a configuration of our router case study with two security enforcement partition instances (2 instances of the Pse partition class), providing at the same time routing and secure forwarding for highly critical domains (via the secu2 instance) and for less critical but more dynamic and more extensible domains (via the secu3 one).

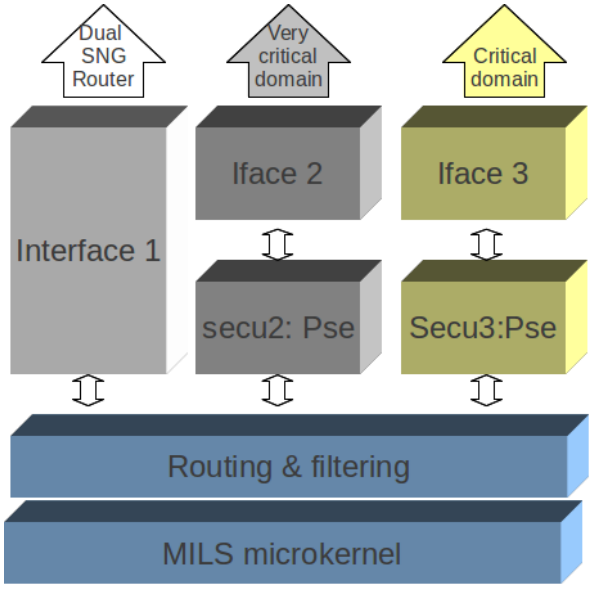

Figure 5 Example of a possible SNG router configuration

The rest of this paper will be focused on the design and validation of the SNG router Security Enforcement Partition (Pse) class. Instances of Pse secure the communication data through dedicated channels. Pse may eventually share a single channel between different network flows, each requiring its own level of safety and security.

Security Enforcement partition design

The Security Enforcement Partition (Pse) must secure the data, conforming to the requirements enumerated in the Protection Profile. Pse is represented as a black box presented figure 6. Pse adds control information to outgoing packets, such as the confidentiality, authenticity and integrity data. This information enables Pse to validate the security of incoming packets. The ESP protocol is the adequate protocol to enforce these security goals and is therefore implemented into the SNG router.

Before securing the communication, Pse must establish the secure channel(s) with the other end entities. To do that, an IKEv2 protocol implementation is included in Pse. When starting or requesting, the SNG router uses static configuration information to invoke the IKEv2 negotiation and then create the secure tunnels.

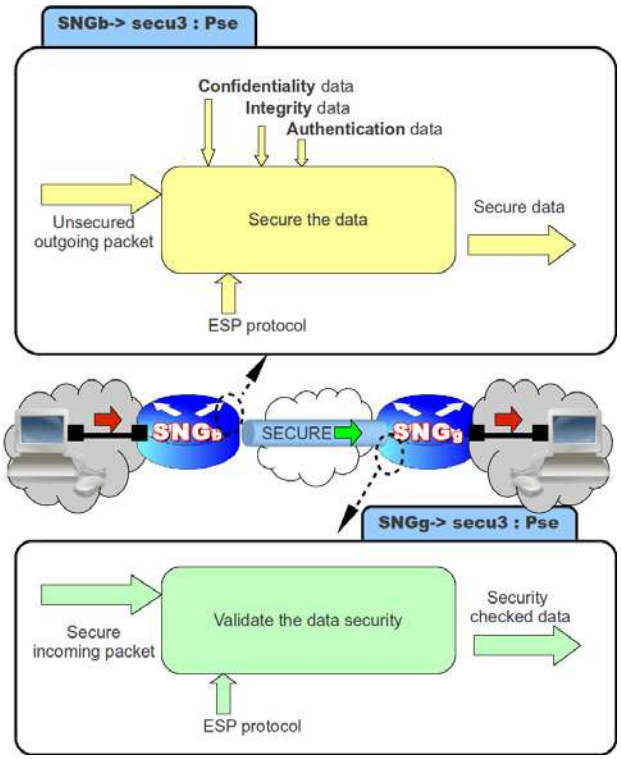

Figure 6 Security Enforcement Partition (Pse) Objectives

The SNG design is based on Simulink [12], a dynamic system modelling tool and on Stateflow [13], a state machine modelling tool. Both are commercial Matlab toolkits. The free and Open Source Gene-Auto [14,15] tool transforms Simulink and Stateflow models into C source code.

The case study of the security enforcement partitions of the SNG router considered in this paper illustrates concretely how the methodology previously introduced can be conducted to generate a safe and secure product from system and state machine models defined during the design step of the development process: the development is conducted from the design of the security partition until the tests on the prototype hardware. For this purpose, Simulink system models and Stateflow state machine models are converted by the GeneAuto transformer into C source code which is compiled to work on an Intel x86 processor (in our case study) with the Sysgo embeddable Real-Time Operating System, PikeOS (MILS and ARINC653 compliant, DAL-A, EAL5+).

IKEv2 designed through Simulink models

The IKEv2 protocol is specified in RfC 5996 [10]. Starting from SNG router requirements, this protocol is implemented for Pse through a graphical model. The model contains a Simulink main part and several Stateflow sub-models to provide the decision and support functions. Figure 7 above shows the decision function currently used to choose if the outgoing packets should be secured or not.

Simulink and Stateflow models enable the developer team to validate the correct behaviour of the implementation as soon as the model has been designed: these tools can stress the models and indicate, for example, if some branches are dead or if the test cases have found boundary errors. Moreover, it eases and shortens the validation and certification tasks, because the graphic representation is more ergonomic and the generated C code is written by tools with qualification evidences. On top of that, Simulink enables the developer team to check the models with formal method-based checking tools, such as Simulink Design Verifier.

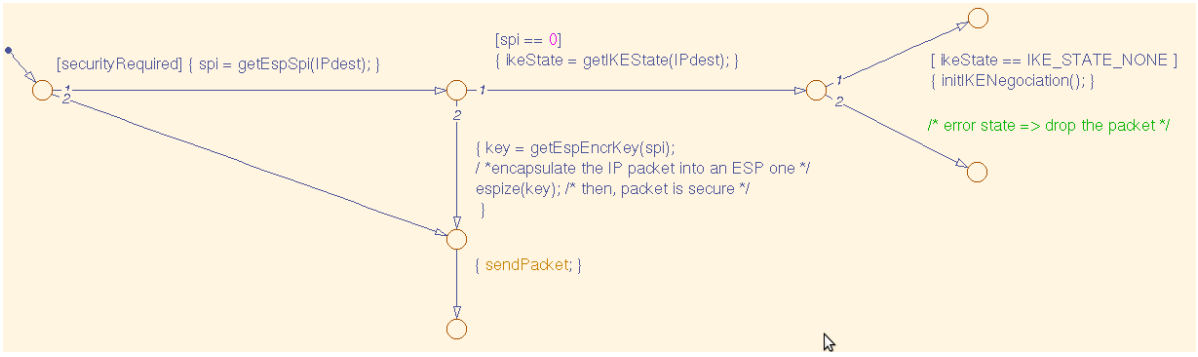

Figure 7 Security decision function

ESP algorithms implemented in legacy C code

Whereas the Simulink and Stateflow models are the core of SNG router source code, the different algorithms, AES256 [16] confidentiality, SHA1 [17] integrity and HMAC [18] authenticity, used to secure the data are not designed through these tools. These algorithms are standardized in documents which already contain a C implementation [19,20]. We reused the C code issued from the Open Source community which has several advantages over a classical design. First of all, this code is widely and successfully used in very different securing applications [21]: websites, firm Virtual Private Networks (VPN)... Thus we can have more confidence in their correctness than in an internal version we might develop. Secondly, by reusing this code we do not have to recode it and we gain time. For the certification, the code we have chosen is very linear. We are working to formally prove its correctness through annotations, such as the ANSI/ISO C Specification Language ACSL [22]. This language enables the verifier to check the boundaries, to verify the termination... Moreover, there are fewer lines of code, compared to the C code generated by GeneAuto from the Simulink and Stateflow models. Another good point is that licenses allow the user to reuse this external code (under conditions acceptable for us). Last but not least, speed of computation is rapid for these algorithms: this code has already been optimized by the community and proven its efficiency.

Supporting configuration for Pse

The IKEv2 protocol needs to be configured to complete establishment of the secure tunnel. Before using the tunnel each end entity must know the other end entity, notably its IP address and the certificate to authenticate the node. The securing mode and other parameters shall be specified too. The SNG router stores this preconfigured data loaded on the installation of the router into the aircraft. This cannot be modified or altered during the aircraft operations. When the router starts, the configuration data block is used to initialize Pse partition.

On one hand, this solution is safer and easier to certify than any dynamic solution and the immutability improves the global security of the partition. On the other hand, this solution is not very scalable because each node must be configured during the installation and re-configured manually when the global security plan is modified. A reconfiguration requires unloading the router, uploading a new file and reinstalling the router. We are working on a protocol to automate security discovery and establishment, reducing the configuration requirements. This protocol is named SCOUT, for “Security Capabilities Over Unsecured Topology”. This work is in progress and out of the scope of this paper.

Statistics about the development

The SNG router Security Enforcement Partition contains an IKEv2 design complemented with 3 algorithms for the ESP protocol and some external libraries the router requires for generation of randomness for cryptographic use. The models contain about 50 Stateflow functions, 2 Stateflow models and 1 Simulink model, converted in 7 files in about 1 minute by Gene-Auto. The 1300 lines of C code are completed by 30 lines of code for the external libraries and 1900 lines of code for the 3 security algorithms, mostly constants and precompiled tables.

Performance evaluation of our SNG router

Environment for Pse testing and validating

Fig 8 presents a basic but instructive use of the SNG router: two embedded systems need to communicate securely with two on-ground systems. A shared link multiplexes the data exchanges between the aircraft in flight and the systems on ground. Moreover, the different communications must be logically segregated. To put this into practice, the shared communication channel is bounded by two SNG routers.

In order to validate our SNG router, we compiled the model with 4 input/output network interfaces[1] and up to 16 route entries in the routing table. This provides up to 16 different and simultaneous secure channel end points.

[1] Note that we need just 3 network devices for the validation topology described in figure 3. We consider one additional network device for future improvements.

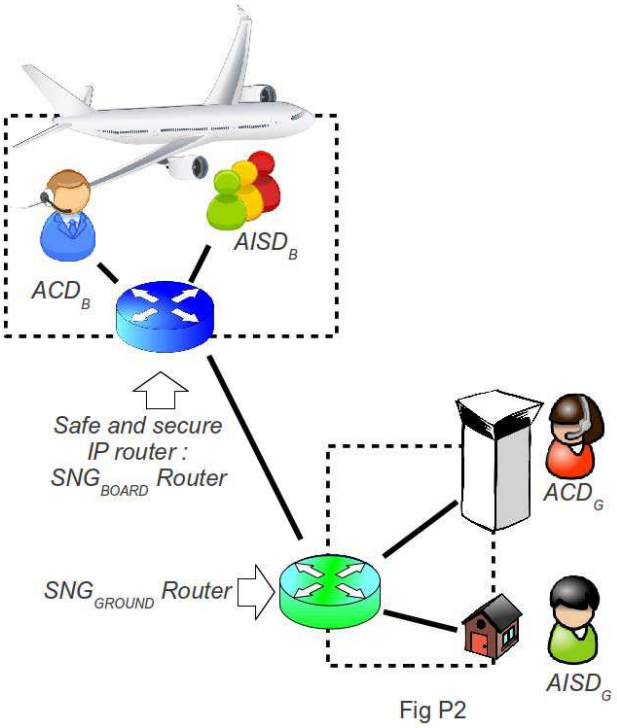

Figure 8 SNG routers to secure aircraft-ground communications

These values are defined in the model by constants and can easily be modified if required, but we did not optimize our router algorithms for high values. 16 routes appear to be large enough for our avionic emulation needs: note that only 4 routes are useful for our experimental configurations (fig 3 & 8).

To secure the data and validate the performances, we compared 4 classes of flows: some flows are not secured, some are encrypted between the routers, some are protected against alterations by inserting integrity and authenticity information and the last class of flows is protected by the whole set of confidentiality, integrity and authenticity security services. To provide the security services, the IKEv2 protocol establishes tunnels between the two SNG routers, and then data are encapsulated into IPv6 packets with ESP extension to provide the security. Unsecured data are forwarded by the routers and bypass the security/encapsulation module.

We used Virtual Machines and the qemu[2] x86 processor emulator to run the performance evaluation. The emulated machine has 32 MB of RAM and a 130 MHz mono-core CPU. Sysgo PikeOS runs our router SNG binary code in several partitions and processes, conforming to Fig 5. A configuration file is loaded when we start the emulator. This file contains the different routes, the security class required to secure each flow, the IP addresses of the SNG router and other miscellaneous parameters.

SNG router maximum additional delay for forwarding and securing packets

We used ICMPv6 Echo request & reply packets of different sizes between 164 and 1364 bytes and measured the round-trip time between the ICMP Echo Request emission and the allocated ICMP Echo Reply reception on a single host. Fig 9 describes our delay measurement experiment: IPv6 packets go through 2 SNG routers and may be secured between them.

If security is needed, a tunnel is established. We measured delays of about 40 ms for the whole IKEv2 negotiation between the routers. During this duration, packets are stored on the initiator router, then secured and sent as soon as the tunnel has been established. We measured the additional delay inserted by the SNG router, and fig 10 summarizes the results.

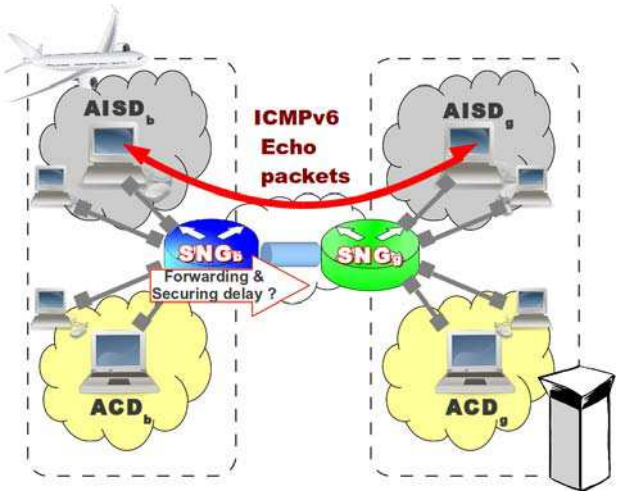

Figure 9 Additional delay measurement

[2] This tool is included in the MILS development framework for Sysgo PikeOS.

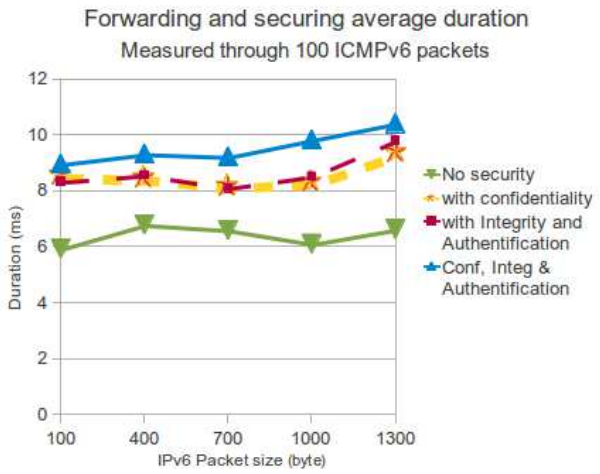

Figure 10 SNG router additional delays

Firstly, fig 10 shows that security services have an impact on the performances: without security, the average forwarding delay is about 6,4 ms, with just confidentiality or just integrity and authenticity, it is increased by 2,2 ms. With the three services simultaneously, forwarding delay is about 9,6 ms, thus 50% longer than without security.

Secondly, fig 10 illustrates that the packet length increases the delay slightly only when security services are enabled. This may be explained by the router design: for safety reason, the design uses only statically-fixed-length buffers of 1514 bytes to perform computations on the IP packets during their forwarding through the different blocs of our SNG router models.

On one hand, the forwarding operations requires the copy of not only the useful packet bytes but all 1514 bytes of the buffers each time a packet is exchanged between two blocs. Thus packet length impact is strongly amortized by the model when forwarding. On the other hand, security services are applied not on the whole 1514 bytes of the buffer, but only on the payload bytes to secure, ignoring the padding; then securing a 100 byte packet requires less computation than securing a 1300 byte packet, whereas forwarding a 100 byte packet requires as much computation as a 1300 byte packet.

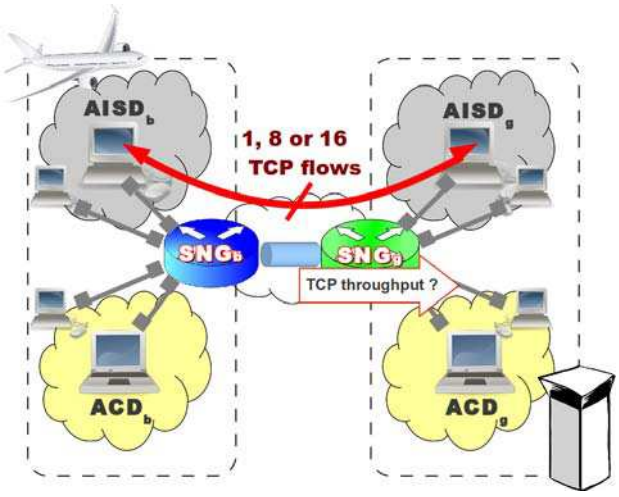

Figure 11 Experiment for measurement of data throughput on a single path

Maximum data throughput through the SNG router

In a second stage, we measured the maximum throughput of packets through our router SNG. We used the iperf command-line program on two end hosts to have a TCP connection between them and to get the maximum throughput for this connection. The TCP congestion control mechanism used on end systems is TCP Cubic [28]. The TCP connection has a satisfactory load increase and converges within a few seconds near the maximum throughput the channel can accept before congestion.

Fig 11 describes this second performance evaluation experiment: the data throughput is measured on one end-system. This node exchanges data with another end-system, through one or multiple TCP connections at the maximum available rate. Two SNG routers are on the path. They may secure the data or not, with the same security services than in the previous experiment.

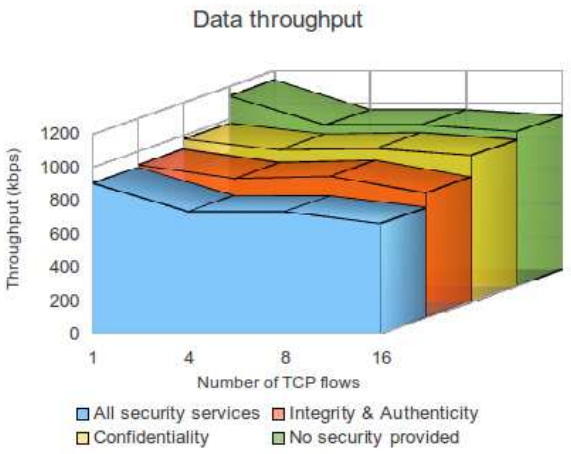

Figure 12 Results for measurement of data throughput on a single path

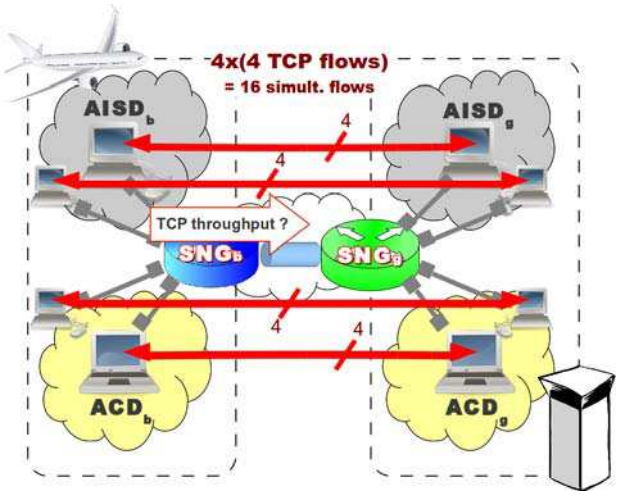

Figure 13 Measurement of data throughput on multiple paths

Fig 12 illustrates the results of our performance evaluation. As previously shown by fig 10, security has an important impact on the performances. Without security, emulators running the SNG router software provide packet forwarding at a rate of about 1150 kbps, while the activation of the three security services decreases this throughput until 665 kbps in the worst case. This figure shows the impact of the TCP flow number: using multiple TCP flows is less effective than having only one TCP flow. This can be explained by the loss bursts appearing because of the congestion of the routers' buffers. These packet losses impact all TCP flows simultaneously, and then TCP congestion avoidance mechanisms of all the simultaneous flows close their congestion windows.

To conclude our experiments, we tuned a third parameter. Whereas the second experiment evaluated the maximum throughput for 16 simultaneous TCP flows on one path, in the third experiment we evaluated the same 16 flows but on 4 different paths, simultaneously used, as illustrated in fig 13. In both cases, flows between the SNG routers are secure against disclosure, alteration and authenticity spoofing. Flows have different extremities but share the same link between the SNG routers. This experiment performs stress testing for SNG implementation of the routing function.

Experimental results are shown fig 14. When we simultaneously use 4 flows per IP address and 4 IP addresses, we find a total throughput of 836 kbps for the 16 flows. This result is above the throughput of 665 kbps for 16 flows on only 1 IP address; this can be explained by the fact we have less flows on each IP, thus TCP congestion avoidance mechanisms are more efficient to maximize the throughput, as shown in the second experiment.

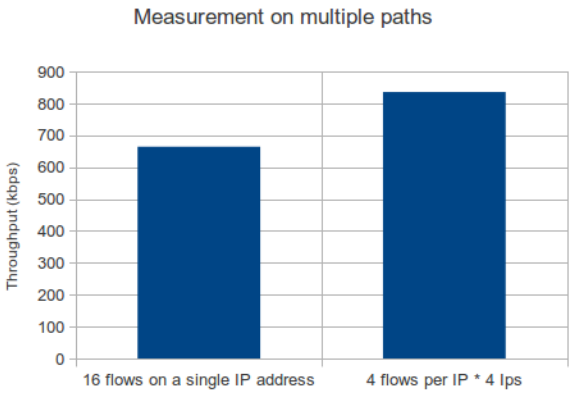

Figure 14 Results for measurement of data throughput on multiple paths

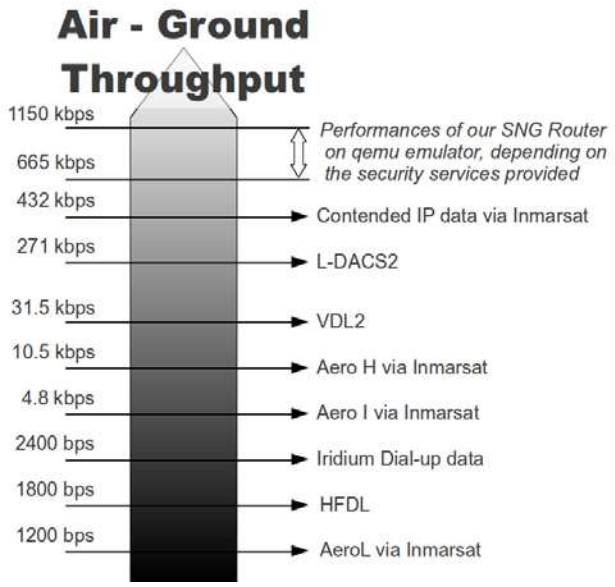

Figure 15 Air-Ground Data link throughput in year 2012

Conclusions of our evaluation

We have demonstrated the ability of our Secure New Generation Router to route and secure data flows between its different network interfaces. Activation of the security mechanisms leads to a reduction in data throughput of about 25%, from 1145 kbps to 905 kbps for a single flow and from 930 to 664 kbps for 16 simultaneous flows. These results are clearly lower than Commercial Off-The-Shelf (COTS) routers for public or private usage, like those used at home or in industry. Furthermore, delays induced by the SNG router on data communication are significant (about 6 to 9 ms per packet, depending on the activated services).

Nevertheless, this router has been designed in order to facilitate certification and evaluation tasks required by the aviation authorities for critical embedded avionic systems. Moreover, the fastest radio-communication data link used commonly in the aeronautical world, now in 2012, enables aircraft to communicate only at a throughput of 432 kbps, as illustrated in fig 15 [25,26,27]. This capacity is much smaller than our SNG router performances, even in the worst case with all security services and multiple flows enabled on multiple network interfaces.

Nonetheless, our overall objectives are to increase data throughput and decrease delays of the final SNG product. Thus, first of all, we plan to use a real target and not an emulator to complete SNG experimental evaluation. Preliminary experiments have shown a significant performance improvement: our real processor runs 20 times faster than the emulated processor, and throughput results should be improved accordingly with the same ratio. Secondly, we are studying some design improvements. For instance, optimizing the configuration of our framework and the development tools we used to make the software should increase significantly SNG router performances.

Benefits around Pse

Security benefits of the IPsec suite

IKEv2 and ESP are two famous protocols mostly used for most VPN. Since years, they bring high-level security services and have already demonstrated their efficiency in general-purpose systems; that is why the SNG router data securing mechanisms are centered on these two protocols. The AES algorithm is nowadays the confidentiality algorithm recommended by different technical specialists, such as the American National Institute of Standards and Technology (NIST [23]). For integrity, the SHA1 algorithm has superseded in 1997 the MD5 as the standard integrity algorithm for most systems [20]. But in 2004 serious flaw were detected, then we would soon change of algorithm and temporary continue to use SHA1 with longer keys to limit the impact. The HMAC algorithm is commonly recognized to be a performing algorithm for authenticity [24]. Moreover, the Open Source implementation we use for these 3 algorithms are often used within the Internet without negative report.

Evaluation benefits

The Common Criteria for Information Technology Security Evaluation is a common standard for security certification of industrial information systems. Its evaluation gives the user a high level of confidence in the security of the successfully evaluated product.

The Security Enforcement Partition has been designed from a high-level software specification completed with a Protection Profile which defines additional requirements to reinforce data securing and the security of the software. This Protection Profile makes easier a future security evaluation of the SNG router: it contains an analysis of

vulnerabilities and the requirements to avoid them and gives some design guidance relative to the router.

The Sysgo PikeOS real-time operating system is another considerable advantage to facilitate SNG router evaluation. This software kernel has already been evaluated up to the level EAL3+. This evaluation could be reused for the Security Target: evaluators will not have to re-evaluate the operating system and can concentrate their efforts on the different security function-related partitions.

To complete SNG router evaluation, some intrusion testing would have to be conducted in the near future.

Safety & Certification benefits

Some SNG router certification tasks are reduced by tool qualification. In fact, the Gene-Auto model transformer into C source code has been designed to be easily qualified. Some qualified C compilers can be used on both the generated and hand-written codes. The DO-178B Design Assurance Level A requires 66 certification tasks to be validated: by using such a methodology, 13 of them are avoided and 3 of them are reduced by the qualification of both the compiler and the model transformer; more details are in [1]. These qualified tools used together enable the developers to provide for the certifiers a maximum of evidence about the safety of the software binaries, reducing drastically the costs and delays of the certification process.

On top of that, formal method-based tools can complete the safety certification: Simulink and Stateflow toolkits enable add-ons to verify and validate the high-level models, such as checking as soon as possible the correct behavior of the models, the absence of overflow...

Conclusions and future work

This paper has presented a Secure New Generation Router for the avionic industry. This software router is designed with safety and security requirements throughout the development. Moreover, it enables the network entities to communicate safely and securely by adding confidentiality, integrity and authenticity to the exchanged data.

The Protection Profile defined for the SNG router contains specific requirements and a dedicated vulnerability analysis. It provides evidence for the security of the software and for the security of the data forwarded by the router. This document can lead to a Security Target to evaluate the final product security level.

The use of model transformers and qualified tools is an improvement for the development and the certification of the SNG router. Some evidence is formally shown and the global process is lighter. To our knowledge, it is the first time the IKEv2 protocol is implemented through Simulink and Stateflow models. This work has been completed by integrating legacy source code into the models.

The generated binary has been validated with the help of an emulator. The performance evaluation we conducted on the SNG router resulted in data throughputs between 650 and 1150 kbps, depending on the security services activated. The router adds a delay between 6 and 9 ms to process the IP packets. These results should be improved soon, with the replacement of the emulator by a real target and by optimization of the general design, of the different models and of the tool chain configuration.

In the near future, dynamic configuration functionalities should complement the SNG router capabilities. Until now, the secure channels are static, and defined in the configuration during the router installation. This may be automated eventually with a certificate managing mechanism and dedicated private key infrastructure. We are currently working on a protocol named “SCOUT” to automate the discovery of Security Capabilities Over Unsecured Topology.

The completion of tools and methods to evaluate our router is another of our research perspectives. We envisage validating the router with other security tests, such as intensive stress testing. This must be conducted conforming to the Protection Profile and the Common Criteria Evaluation methodology. Moreover, we project to use annotations to enforce safety confidence in the C source code. This could be conducted first on the legacy source code for the ESP algorithms, then on the generated source code.

References

[1] New methodology to develop certified safe and secure aeronautical software – an embedded router case study, Antoine Varet, Nicolas Larrieu, 30th Digital Avionics System Conference

[2] S. Kent and K. Seo. Security Architecture for the Internet Protocol. RFC 4301 (Proposed Standard), December 2005

[3] RTCA SC-167, EUROCAE WG-12, 1992, DO-178B Software Considerations in Airborne Systems and Equipment Certification, Washington, DC, RTCA, Inc.

[4] Common Criteria for Information Technology Security Evaluation, version 3.1, ISO/IEC 15408, http://www.niap-ccevs.org/cc-scheme/cc_docs/, July 2009

[5] The WindRiver website, http://www.windriver.com/products/vxworks/, January 2012

[6] PikeOS embedded Virtualization, SYSGO AG, Germany, http://www.sysgo.com/products/pikeos-rtos-and-virtualization/embedded-virtualization/, December 2011

[7] T. Ylonen and C. Lonvick. The Secure Shell (SSH) Protocol Architecture. RFC 4251 (Proposed Standard), January 2006

[8] IEEE 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements, IEEE Standards Association, July 2004

[9] D. Harkins and D. Carrel. The Internet Key Exchange (IKE). RFC 2409 (Proposed Standard), November 1998. Obsoleted by RFC 4306, updated by RFC 4109

[10] C. Kaufman, P. Hoffman, Y. Nir, P. Eronen. Internet Key Exchange (IKEv2) Protocol. RFC 5996 (Proposed Standard), September 2010. Updated by RFC 5998

[11] S. Kent. IP Encapsulating Security Payload (ESP). RFC 4303 (Proposed Standard), December 2005

[12] Simulink, The MathWorks Inc., http://www.mathworks.com/products/simulink/, January 2012

[13] Stateflow, The MathWorks Inc., http://www.mathworks.com/products/stateflow/, January 2012

[14] Toom, A., et al., 2008, Gene-Auto: an Automatic Code Generator for a safe subset of Simulink/Stateflow and Scicos, 4th European Congress ERTS Embedded Real Time Software

[15] Rugina A., et al., 2008, Gene-Auto: Automatic Software Code Generation for Real-Time Embedded Systems, Data Systems in Aerospace (DASIA) 2008

[16] Specification for the Advanced Encryption Standard (AES) Federal Information Processing Standards Publication 197 (FIPS-197), 26/11/2001, National Institute of Standards and Technology (NIST)

[17] Secure Hash Standard (SHS) Federal Information Processing Standards Publication 180-2 (FIPS 180-2), 25/02/2004, National Institude of Standards and Technology (NIST)

[18] The Keyed-Hash Message Authentication Code (HMAC) Federal Information Processing Standards Publication 198a (FIPS-198a), 06/03/2002, National Institute of Standards and Technology (NIST)

[19] D. Eastlake, US Secure Hash Algorithm 1 (SHA1), RfC 3174 (Informational), IETF, September 2001, Updated by RfC 4634 & RfC 6234

[20] H. Krawczyk, M. Bellare, R. Canetti, HMAC: Keyed-Hashing for Message Authentication, RfC 2104 (Informational), IETF, February 1997, Updated by RfC 6151

[21] OpenSSL Usage Trends, The BuildWith Technology Usage Statistics Web and Internet Technology Usage Statistics, http://trends.builtwith.com/Server/OpenSSL, 05/12/2011

[22] The ANSI/ISO C Specification Language (ACSL), http://frama-c.com/acsl.html, 05/12/2011

[23] Lynn Hathaway, "National Policy on the Use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information" (PDF), June 2003, retrieved 2011-02-15

[24] D. Eastlake, T. Hansen, US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF), RfC 6234 (Informational), IETF, May 2011

[25] Inmarsat website, www.inmarsat.com/Services/Aeronautical, March 2012

[26] The Eurocontrol Website, Working together to deliver the Single European Sky, www.eurocontrol.int/vdl2/, March 2012

[27] The RadioReference.com, wiki.radioreference.com, March 2012

[28] Injong Rhee, and Lisong Xu, CUBIC: A New TCP-Friendly High-Speed TCP Variant, ACM SIGOPS Operating Systems Review - Research and developments in the Linux kernel, Vol 42 Issue 5, July 2008 P64-74

Acknowledgements

We would like to thank Rupert Salmon and John Kennedy for their help in editing this paper. We express also our gratitude to Julien Marchand for his excellent help to design and implement SNG router IKEv2 protocol.

Email Addresses

For further information, Antoine Varet can be contacted at Antoine.varet@recherche.enac.fr and Nicolas Larrieu at Nicolas.larrieu@enac.fr.

2012 Integrated Communications Navigation and Surveillance (ICNS) Conference

April 24-26, 2012