Publishable Summary

Project number:	318353
Project acronym:	EURO-MILS
Project title:	EURO-MILS: Secure European Virtualisation for Trustworthy Applications in Critical Domains
Start date of the project:	1st October, 2012
Duration:	42 months
Programme:	FP7/2007-2013

Date of the reference Annex I:	8th July, 2015 (V3)
Periodic report:	Publishable Summary (as part of D42.4 “3rd periodic report according to EC regulations of the model contract”)
Period covered:	01.10.2014 – 31.03.2016
Activities contributing:	All
Due date:	May 2016
Actual submission date:	14th July, 2016 (V1.1)

Project Coordinator:	Dr. Klaus-Michael Koch Technikon Forschungs- und Planungsgesellschaft mbH (TEC)
Tel:	+43 4242 233 55
Fax:	+43 4242 233 55 77
E-mail:	coordination@euromils.eu
Project website:	www.euromils.eu

Chapter 1 Publishable Summary

Project name: **EURO-MILS**
Grant Agreement: **318353**
Project website: http://www.euromils.eu/
Contact: coordination@euromils.eu

Start date: 1st October 2012
Duration: 42 months

The mission of EURO-MILS *was to develop a solution for virtualisation of heterogeneous resources and provide strong guarantees for isolation of resources by means of Common Criteria certification with usage of formal methods.*

The main goals of the EURO-MILS project were to develop market relevant technologies and concepts for virtualisation of heterogeneous (embedded) systems and the formal verification for those systems as part of rigorous cross-European security certification.

Motivation: Based on embedded systems, cyber-physical networks are part of our society, and gain wider spread and importance. Next generations of aircraft and cars will be tightly interconnected with each other, with the internet, and other infrastructures. The same holds for many industries and areas of our life such as healthcare, energy, finance and mobile communication. Non-secured network devices can be hacked and exploited to affect their functionality, change control, or steal specific information. In order to provide secure and safe trustworthiness and exclude devastating, unauthorized use of critical systems, to control access in an organized and certifiable fashion, the EURO-MILS project has introduced into the European trustworthy ICT landscape a verified and design-validated MILS platform: a small virtualisation platform that offers the secure decomposition of complex embedded systems into independent components.

Objectives & Overall Strategy: To address the problem of trustworthiness, we introduced the certified MILS platform into the ecosystem of European trustworthy ICT. The EURO-MILS platform fulfils the following requirements:

- Fit the technological, business, and legal environments
- Generates trust by design – the EURO-MILS platform allows the composition of complex trustworthy systems following the MILS approach
- Generates trust by high-assurance – the EURO-MILS platform went through a computer-supported verification ("formal methods") as well as a strong human validation (CC security standard certification)
- Is strongly aligned with European industrial needs and two prototypes in avionics and automotive have been co-developed to the MILS platform.

The strategy of the EURO-MILS project was built on four independently managed activities (A1-A4) and eleven tightly integrated work packages (WP):

- A1 "Business and Legal Foundations for Trustworthy ICT" provided a solid foundation for the project that consisted of industrial requirements, certification requirements, as well as business impacts and legal implications.
- The focus of A2 "Trustworthy Design by MILS" was the development of the MILS technology as the base for trustworthy designs and its applications on the use cases from avionics and automotive defined in A1, including the developments of an avionics and automotive prototype.
- A3 "Assurance for End-Users" focused on assurance techniques for end users comprising certification requirements from A2, usage of the CC standard for high-assurance security evaluation including formal methods, and providing a cross-European high-assurance security evaluation methodology.

- A4 “Programme Management and Dissemination” wrapped the project by focusing on standardisation, dissemination and management activities.

Description of the work performed and results since the beginning of the project

The EURO-MILS project started in October 2012 and ended after 42 months in March 2016. During the first project year, the focus was placed on the analysis of industrial and certification requirements. During the second project year, the focus was to initiate and support works on all project topics including defining MILS architectures and components, implementing defined elements and implementing execution environment, formal modelling, common criteria evaluation as well as creating a base for the out-of-the-project MILS community and MILS standardisation. Within the third project period (M25-M42) the prototypes have integrated the components to full MILS architectures in the prototypes, and successfully tested it. The MILS platform based on PikeOS was analysed by the introspection tool ramooflax and specific parts validated by formal proofs and formal-model generated tests. The MILS platform behaviour and selected security properties were formally modelled and proven and CC-assessed according CC evaluation scheme. The insights were generalised into high-assurance methodology.

The objective of **WP11 (Industrial Requirements)** was the collection of a set of industrial requirements for virtualisation of resources. A specified set of requirements helps identifying system components, interfaces and responsibilities between components. These requirements helped to define a high-level architecture for the prototypes developed in EURO-MILS project. One of the focus areas of D11.1 was to clearly define those requirements, characteristics, and the scope of the virtualisation platform, which is to provide assurance, integrity, and security as the base for the trustworthiness for critical embedded systems with scarce resources. Moreover, the document should ease formal specifications. Based on an analysis of available assurance techniques, a security target (ST) for PikeOS as a MILS separation kernel has been prepared within **WP12 (Certification Requirements)**. The ST is intended to be evaluated in accordance to Common Criteria both in France and in Germany. The assurance level chosen was EAL 5+ with extensions that support the robust and reliable separation of partitions. Based on the security target, a draft of a protection profile (PP) for MILS separation kernel has been prepared, its assurance level is EAL 5+ as well. The ST and the PP have been used as input for evaluation activities, which lead to an improvement of both documents with a more precise and more adequate formulation of the requirements. In particular, some security objectives are achieved by a close collaboration of the TOE with mechanisms implemented outside the TOE border (hardware, software in userland). Special care was required in formulating the security function requirements to cover exactly the contribution of the TOE in achieving the objectives. The ST and the PP have been submitted to both the French and the German CC certification authorities (ANSSI and BSI). Use cases from avionics and the automotive field have been investigated and domain specific functional security requirements have been produced for them. These requirements have been formulated such that they meet the demands of the CC composite evaluation approach (WP21) with a separation kernel complying the ST (and the PP).

The objective of **WP13 (Business, Legal and Social Acceptance)** is to analyse the business impact of such a trustworthy technology in markets adjacent to the core automotive and avionics targets. It studies the potential of the EURO-MILS platform in markets such as healthcare, finance, smart home, mobile communication, etc. During the first period, we worked first on the common definition of trustworthiness, virtualization, security, and safety. They are the starting point of our work. Defining the project terminology and glossary allows members of the EURO-MILS project to discover and to reduce potential ambiguities and to ensure a consistent, complete and common understanding of the terms. A document was published internally and was included in WP13 final deliverable (D13.2 MILS for business, legal and social domains) in month 36. We also contacted an informal industry panel, made up of professionals in different industries where the EURO-MILS technology could make sense (medical, finance, energy…). We have contacted 245 professionals, received 72 direct answers and made 39 interviews on the following themes: security and safety, virtualisation and partitioning, certification and user assurance. To guide the interviews, we created a questionnaire. The idea behind the questionnaire is to understand the value of the EUROMILS deliverables in adjacent markets. We analyed the answers to setup guidelines to the project. We prepared an end-user survey to understand how end-users valuate security and safety in the products they buy. During the second period, we analysed the Industry panel answers and documented them into the WP13 deliverable. We also ran the end-user survey in six geographies all across Europe including Germany, United Kingdom, France, Spain, Italy, and Benelux. This consumer survey theme was about analyzing the social value of secure products and understanding the user assurance mechanisms. We received more than 500 answers and started the analysis. Finally, we published a draft of the final document to the European Commission. During the third period, we continued the analysis of the survey. Security is important for consumers and is linked to privacy but implications are not very well understood. Therefore, consumers would rely on a security label or evaluation to ensure the device they are buying has the right level of security. Finally, we ran a Big data analysis to listen to the consumers conversations on security when discussing about smartphone purchases. The objectives were to supplement our existing work with social media and consumer information data sources to gain direct customer insights. Smartphone security is not the main discussion theme, but features and price are most discussed. With all these analysis and results, we published the final document to the European Commission.

The first scope of **WP21 (MILS Architecture)** was to establish a common understanding between partners on what a MILS architecture means and what are its components given the absence of a standardized view. Then it served to define a common architecture for the prototypes. A workshop on PikeOS was held in Mainz in cooperation with WP22 (see WP22). As first deliverable (D21.1), by many teleconferences, SVN repository interactions, and a two-day workshop at AOS in Toulouse, the partners have developed a common view on “MILS architecture” consisting of an introduction to the MILS background coming from both security and safety, a common view of a MILS architecture template that can be applied to individual MILS systems instantiating the template and a set of definitions, defining, for example, MILS components, resources, security policies, partitions. One of the insights gained was that software and hardware components are best treated as having equal status, thus a chapter on components lists both software (separation kernel, generic device abstraction component, console system component, network system component, file system component, audit system component, generic application component) and hardware components (processing unit, memory management units, input/output memory management units, I/O sharing, timers).

Partners also received and acted upon some feedback from Rance DeLong of Open Group who viewed parts of an early version of the text. D21.1 has been delivered at M12. The next step of WP21 was to apply the MILS architectural template to the EURO-MILS demonstrators in automotive and avionics, which means that components were located with regards to the EURO-MILS architectural template (e.g. MILS system, MILS platform and MILS core). Moreover WP11 requirements have been assigned to individual components. This work has been documented in D21.2 “MILS architecture for avionics and automotive” (delivered at M15). Based on a comparison worked out by the subcontractor Fraunhofer IESE, on the two prevalent Common Criteria methods for Compositional Certification, Composed Assurance Package (CAP), and Composite Product Evaluation according to CCDB (CEP), partners first established a compositional certification methodology based on “*a posteriori*” arguments done at the composition stage (D21.3). Next, EADS IW and OPSYN established a security problem definition (threats, attackers, security objectives) for the automotive and avionics prototypes. Applying the D21.3 methodology has led to the insight that, for a MILS setting, it is feasible to establish some security properties earlier than at composition stage, but rather at separation kernel certification stage (“a priori” arguments). From this, a new evaluation methodology has been evolved, which is documented in WP33.

Based on our experience with T2.1.1 and WP31, the partners agreed that the most interesting security property to formally model was non-interference. First, TUE did proofs about the relation between different non-interference models (Rushby, GWV, and a property derived from Van der Meyden http://dx.doi.org/10.5281/zenodo.47983), which is relevant for model acceptance because these are the known published specification options. An approach was developed that allows the expression of local security properties of applications and show that they remain valid within a system with a separation kernel. The separation properties of the kernel required are compatible to those established in WP31 (by CISK with bisimulations and MCISK), and, with support from DFKI, TUE has produced a formal (Isabelle/HOL) instantiation of the firewall, where the firewall based on a CISK separation kernel is applied to either individual applications or an application within a partitions (“firewall system” and “domains programs” in http://dx.doi.org/10.5281/zenodo.47980). A high-level proof how to use MCISK instead CISK also has been developed.

WP22’s (MILS Components) goal was to research, design and implement the infrastructure defined in WP21, which was required by the automotive and avionics prototypes of WP23. This means that PikeOS was needed to be ported to the chosen hardware platforms and extended with support for hardware-based I/O virtualisation. We started with a well-attended (20+ participants from 11 partners) workshop organised by SYSGO in Mainz (Germany) on the PikeOS separation kernel. It covered all aspects of the development process, from setting up the basic environment to finding a security issue in a device driver. The first step into getting PikeOS to run on a new platform consisted of creating a so-called Platform Support Package (PSP). This PSP enables the PikeOS separation kernel to use a number of basic facilities of the underlying hardware, such as a timer, interrupt vectors and memory mapping. The PikeOS support for the avionics platform was finished in M14, while the support for the automotive platform was provided to project partners only in M22. The reason for the latter delay was due to repeated failure by a project-external hardware supplier to provide correct and complete specifications, necessitating yet another platform change around M15.

Once the platforms were decided upon in WP12, in parallel with the aforementioned work we also started investigating the I/O virtualisation capabilities of the selected hardware. We first created a comparative overview of these properties, tabulating the properties of the various I/O memory management units (IOMMUs). This overview was amended every time we had to switch platforms for the automotive prototype. Next, based on the existing IOMMU support for x86 in PikeOS, we created an IOMMU interface and implementation for the avionics platform. This experience, combined with the requirements analysis of the automotive prototype and capabilities analysis of its IOMMU, has lead to the design of a generic I/O virtualisation interface for PikeOS. Work on the implementation of the IOMMU support for the automotive platform incurred delays due to the aforementioned issues with the hardware supplier. Fortunately, in M25 we finally received all necessary documentation and sample code, after which the work went well. In M24-M30, we successfully implemented the IOMMU driver for the automotive platform.

WP23 (Prototype Integration) targeted several major topics: development and integration of the prototypes and the prototypes test-bed for the avionics and automotive domains, respectively, together with their validation. In both domains, we have worked on the prototype platform to integrate supporting software from WP22 to enable the security features each platform offers. At the same time, test concepts for each domain have been developed, and the respective test-beds have been successfully implemented. In the avionics domain, the prototype software modules have been developed and successfully integrated on top of PikeOS running on the P4080 based hardware. For the automotive prototype, PikeOS has been extended to support the hardware virtualization and the firewall features of the Texas Instrument Jacinto 6. The integration of the automotive prototype architecture on top of PikeOS running on TI Jacinto6 has been successfully achieved.

Three test-beds have been developed:

1. MILS platform based on PikeOS testbed - to test PikeOS behaviour using the hypervisor Ramooflax as introspection tool
2. Avionic testbed - based on unit tests of prototype components and Scapy tool to validate components implementation and prototype integration from functional and security point of view
3. Automotive testbed - partially integrated in a larger automatic test environment consisting of continuous integration and functional test.

The final avionic prototype fully implements 88% and partially 12% of the initial requirements, the automotive prototype 72% and 15% respectively. The reasons why some requirements were partially or not implemented for the avionic prototype are detailed in D23.1 and in D23.2 for the automotive prototype. Almost all implemented requirements were validated by testing activity; the details of the prototypes validation are shown in D23.4. The validation report shows that the MILS architecture formed a solid basis for the implementation of the avionic and automotive use cases and the majority of executed tests show that the security features have been correctly implemented.

During the first year of the project, the main work within **WP31 (Assurance by Formal Methods)** has been towards a generic model of Separation Kernels and the demonstration that parts of PikeOS are instantiations of this model. At the end of the first period, such a generic model is described in Deliverable D31.1 and one system call (IPC) of PikeOS is proved to be an instantiation of this model. A paper was prepared describing this result. This new model provides a model of separation with interrupts and control, two aspects needed in practical applications like PikeOS. These aspects were missing in all existing models. To achieve this result, a deeper understanding of separation properties was necessary. A thorough literature study shows that two formulations dominate. The Greve, Wilding, and Vanfleet model and the model proposed by Rushby. A paper was prepared describing the conclusion of this comparison of information flow policies. Beside a comparison between the two main models, a conclusion of the literature study was that out of-the-box models were not directly applicable to PikeOS and a new model was needed.

During the second year, advances in WP31 have been made in both the generic model and the implementation model. Additionally, we have identified implicit assumptions behind the model, i.e., the assumptions that are not formulated in Isabelle but which have been made during modelling. We have published a paper about that topic. Two major aspects have been added to the generic model. First, it has been given an explicit notion of time. This allows proving both time and space separation. The unwinding methodology has been extended with new proof obligations that ensure time separation. We have finished a draft version of a multicore model (http://www.zenodo.org/record/48658). Currently, we are investigating whether multicore PikeOS satisfies the assumptions made for the multicore model. All relevant PikeOS API calls have been modelled in the implementation model. Besides the IPC, the event API, the memory, ports, external file providers and user locks have been modelled. For all these models, all necessary proof obligations have been discharged. In other words, all of these actions have been proving to ensure time and space separation. A paper on this topic has been published at the NASA formal methods conference.

Year 3 focused on two parts: first, the formal model has been refined and improved. It has been studied whether it is possible to formulate the entire generic model as a series of monadic transformations. We start with a Mealy machine and provide transformations that add time, aborting and waiting behaviour, and so on. Each of the transformations is shown to preserve intransitive non-interference. On this topic, a paper will be written. The second part focused on how the formal model is to be used in a certification context: what is its role in, e.g., an EAL5+ certification.

WP32 (Common Criteria Security Evaluation) dealt mainly with the formal evaluation process according to Common Criteria (CC). A high evaluation assurance level EAL5+ was chosen for the separation kernel instantiated through PikeOS. The aim of a security evaluation is to provide trustworthiness to the end-users.

Partner SYSGO created and provided CC developer’s contributions to TSYS. The evaluation facility of TSYS performed the evaluation process for all the assurance components included in the assurance package EAL5+ according to the CC evaluation methodology as required within German national CC certification scheme by BSI. Thereby, TSYS used the developer’s CC contributions, PikeOS and test environment installations supported by SYSGO and its own laboratory environment and an additional tool prototype, which was set-up to support CC vulnerability analysis. This tool prototype was developed by EADS F IW and was implemented together with TSYS in Bonn yielding some preliminary results for AVA.

The results of this evaluation process are documented in the related evaluation technical report in a detailed way as required by CC.

The complete evaluation technical report (ETR) was provided to BSI, whereby a specific evaluation technical report (I-ETR) was communicated to ANSSI by the partner TCS. TCS also synchronized the evaluation progress with ANSSI and, particularly, all the evaluation results from the certification in Germany were communicated and discussed between TCS and ANSSI. The project deliverable version of the evaluation technical report was compiled for reporting the evaluation effort to the partners and to the EU commission.

The formal CC evaluation process also yielded additional knowledge on the preparation of developer’s contributions and evaluation of separation kernel as product type.

Outside of CC, WP32 also comprised assessment of formal model created by other project partners in form of a “CC developer statement”. This independent assessment of the formal model, also with respect to the fulfilment of the related CC requirements, was performed by TSYS, DFKI and TUE.

WP33 (Cross-European Certification) dealt with assurance methodology for high EAL (above level 4) and with composition. During workshops, the challenge of high assurance (above EAL4 where recognition between countries is complex) and of composition have been introduced to all partners. The JIL approach has been followed. Therefore, the deliverable D33.1 deals with “Application of Attack Potential to MILS” and also with “Attack methods for MILS” in order to follow the approach adopted for other technologies. Furthermore, a draft version of “composition for software platform” has been released in order to address the question of composition on top of software platform.

Within ***WP41 (Dissemination, Standardisation and Exploitation)***, the early established robust IT infrastructure (web site, SVN repository including web access, mailing lists including mailing list archives) was updated regularly. EURO-MILS has also been advertised by web pages, press releases and internal partners’ newsletters. Hardcopies of the EURO-MILS project flyers have been distributed by partners at various events. EURO-MILS organised peer-reviewed MILS workshops (http://mils-workshop-2015.euromils.eu/ and http://mils-workshop-2016.euromils.eu/) and a forum for the MILS Community (http://mils-community.euromils.eu/). MILS workshop papers and EURO-MILS public deliverables have been archived at ZENODO (https://zenodo.org/collection/user-mils). The project is visible on twitter and LinkedIn. Newsletters have been published and distributed, amongst others, to industrial partners contacted in the context of the project results. A list of dissemination activities has been compiled and updated periodically. All details regarding dissemination, exploitation and standardisation activities can be found in D41.3.

WP42 (Project Management & Activity Coordination) was responsible for the effective organization of the project and covered all relevant management components, including risk and innovation management.

Final results and their potential impact and use

The technology and the framework that has been developed within the EURO-MILS project aims at providing a technical infrastructure for building generic and secure virtualisation solutions while providing high guarantees of the claimed functionalities. The emerging technology of trustworthy virtualisation solutions is a rapidly growing market. Today, virtualisation has left mainframes and servers, spread across home and office computers, and have reached concurrent embedded systems. Moreover, analyses (e.g. Balacco et al., “Virtualization for Mobile & Embedded Systems”) show that the trend goes that the hypervisor will become the operating system of the future, and the guest OS’es will be reduced to user level run time systems managing the processes, and designed to run on a hypervisor, e.g. current research by Microsoft on Hyper-V. To be at the forefront, it is important to invest in embedded virtualisation, as this will give Europe a competitive advantage in future OS’es too. The impact of security and safety of these systems will have a significant influence on how fast European companies will reach the market. Furthermore, as several server virtualisation solutions were initially developed in Europe too, EURO-MILS is also continuing a tradition. The MILS approach not only has lowered costs by protecting computing systems from costly malicious security attacks and accidental errors, it also will enable and drive entirely new business models. In recent years, the economic impact of security attacks (e.g. attacks of virus Stuxnet on Iranian SCADA systems) has shown that we need a more setup-friendly and secure environment to run applications provided by third parties. EURO-MILS has worked out a key technology for enabling new business models with a high economic impact, e.g. running trusted, non-trusted, and legacy software tightly integrated under control of a certified MILS platform and shown its viability in two prototypes that use virtualisation by the MILS platform (avionics and automotive). EURO-MILS has shown up ways for formal modelling and CC evaluation of separation kernels. The project has empowered the European software and specifically virtualisation and integration business and its competitiveness not only by supporting a highly innovative technology, but also by bringing together the essential European players, be it research, industry or SME.

The EURO-MILS Consortium: The consortium brought together 15 partners from 5 different countries: reputable universities and recognised companies from five European Union member states (Austria, Netherlands, Germany, France, and Belgium). All partners are experts in their field. This partnership of experienced professionals resulted in a successful project.