SYSGO
EMBEDDING INNOVATIONS

Security by Design: Introduction to MILS

MILS Workshop

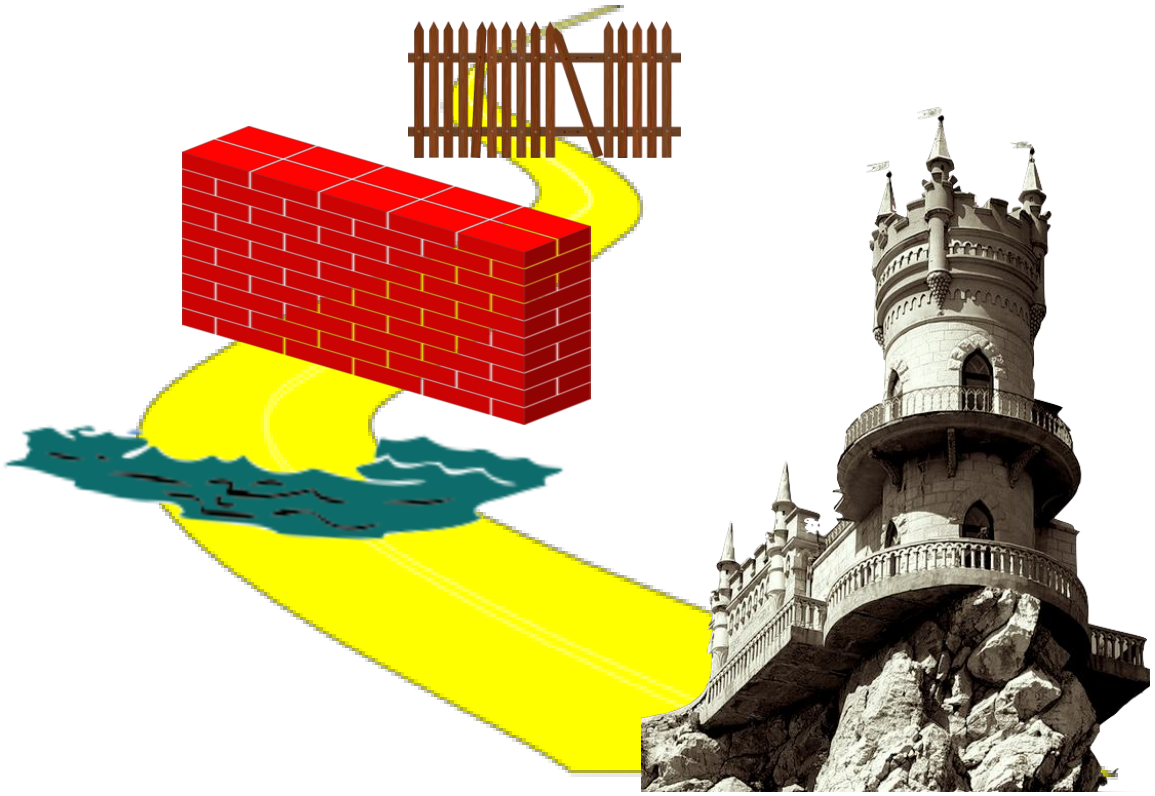
Embedded World Conference 2017

Author(s): Dr. Sergey Tverdyshev, Director R&T
Date: 14.03.2017
Version 1.0

SECURING ASSETS:
EARLY DAYS AND NOWADAYS

Protecting Assets

- **People started protecting their assets (e.g. life) from the very beginning of their existence**
- **People started building**
  - Fences
  - Walls
  - Trenches + water
  - Air-gapping
  - …



SYSGO
EMBEDDING INNOVATIONS

Protecting Assets

- **People started protec**
  beginning of their ex
- **People started buildi**
  - Fences
  - Walls
  - Trenches + water
  - Air-gapping
  - …

Fences + Walls + Air-gap + Underground + …

YOUR NEXT LIVING-WORKING PLACE?

or embrace information flow?

What is going to be hacked?

Short Answer: the weakest link

Long Answer: Attack Surface

- **Typically attacks aim at**
  - components with the exposed interfaces
  - information flow within system, i.e. components interaction
- **Thus, the attack surface is the full system architecture**
- **Security is the integral system property!**
- **Without a clean design, it is extremly difficult to identify/define the attack surface**

Example: Hacking Cryptography

- **Hacking cryptography often imagined as using BIG supercomputers, hacking crypto algorithms, hacking crypto protocols ….**
- **In reality, hacking crypto is a hard task**
  - unless some secret services have placed backddors ☺
- **In reality, it is easier to attack how the crytpo subsystem/engine is integrated in the system**
  - Integrated means: information exchange between security domains, calling APIs, storing (critical) auxilary data
- **Example: Talk ”Crypto wont save you either” by Peter Gutmann**
  - List a lot of prominent hacks, for all of them crakcing crypto was **not** necessary
  - All of them targeted **integration**

SAFETY AND SECURITY

Safety and Security

- **Safety – system shall not harm the environment**
  - Example in aircrafts/cars: passengers shall stay alive and unharmed while transportation from start to destination
    - System: aircraft/car
    - Environment: passengers
    - Harm: crash leading to deaths

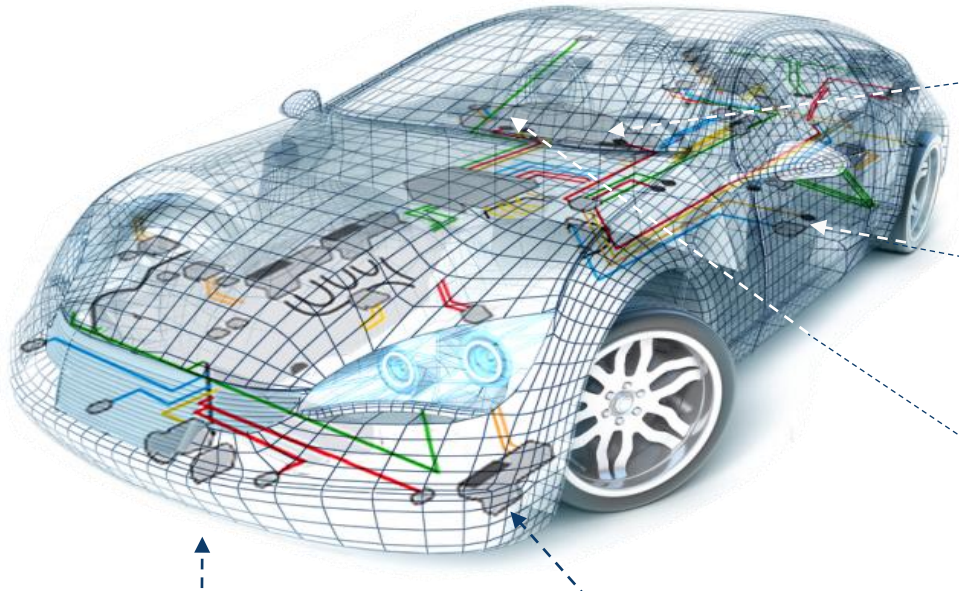
- **Security – environment shall not be capable to harm system**
  - Example in information gateways: information shall only be read/written by authorized subjects
    - System: information processing device
    - Environment: unauthorized subjects (hackers)
    - Harm: modification or leak

SYSGO
EMBEDDING INNOVATIONS

Aircraft Today

- **Aircraft is network based (AFDX & IP)**
- **Increasing usage of common computing resources**
  - IMA, Open World
- **Open World domain with COTS software**
  - Wi-fi products, Linux
- **New IT services**
  - Pilots (tablets), passengers, crew, maintenance
- **Increasing integration and information flows between systems**
- **Aircraft is heavily connected to other IT services**
  - Airlines, ATC
- **Aircraft is connected to INTERNET**

Highly integrated ECUs with COTS SW

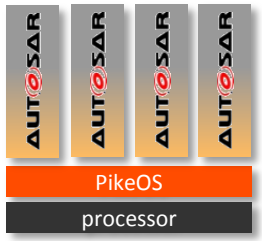


Instrument Cluster

Connectivity Box

Infotainment Head Unit

Domain Controllers



Driver Assistance Systems

Tachograph

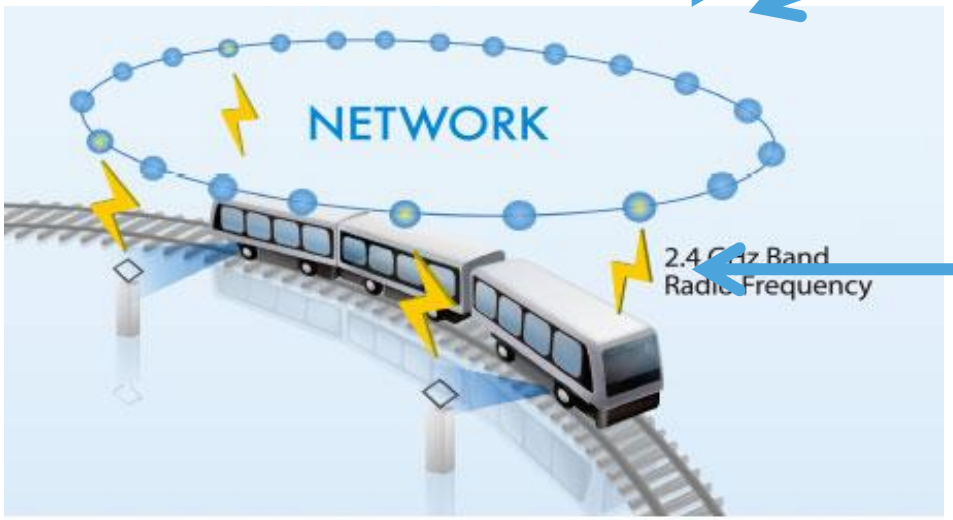
SYSGO
EMBEDDING INNOVATIONS

Railway going online

Field elements

Wired,
public wireless

Interlocking



CBTC (Communications-Based Train Control)

Operation Management and Supervision Systems

SYSGO
EMBEDDING INNOVATIONS

Common Challenges in Cyber-Physical-Systems

- **Functionality density is increasing**
  - Integrate functions on small numbers of ECU
  - Reduce the number of ECUs or keep (at least) the same
  - Benefit on powerful COTS HW and SW
  - **Need proper se**

Affordable assurance

- **Heterogeneous information flows**
  - Systems are interconnected and e ed to exter l world
  - Usage of common network infr
  - **Need proper separation a ontrol of information flows**

- **High-assurance for mixed-critica CUs**
  - Functionalities have different assurance requirements, e.g. safety vs. security
  - The overall assurance design shall be enough to run the most demanding one
  - **Need proper compositional certification approach**

SYSGO EMBEDDING INNOVATIONS

Secure Design Methodology for integrated CPS
Sharing Challenges

Challenge: Resources sharing

- **Resources**
  - CPUs
  - Memory, IO memory
  - Flies, drivers, devices, buses
- **Safety**
  - Integrity, availability
  - Isolation, application errors, fail safe
- **Security**
  - Integrity, availability, confidentiality
  - Possible side channels via shared resources
  - Resources and API are attack surface

MILS methodology addresses

Resource Partitioning

Challenge: Time sharing

- **Time**
  - CPU cycles
  - Time effects of accessing shared resources, e.g. buses
- **Safety**
  - Availability, deterministic behavior, meeting deadlines
  - Right balance between time- and event-triggered tasks
- **Security**
  - Availability, confidentiality
  - Possible timing side channels via shared resources, e.g. caches, busses
  - Time is the attack surface

MILS methodology addresses

Time Partitioning

SYSGO EMBEDDING INNOVATIONS

Common: Assurance via Certification

	Aerospace	Automotive	Railway	Smart Grid
Safety	Long history of standardisation	Recently introduced standard	Long history of standardisation	Based on industrial automation
Security	On-going work on security standard	Staring work on security standard	Staring work on security standard	Many national initiatives; defining path

Standards are focused on
development processes, risk modelling, V&V, and domain specifics

MILS – is architectural principle addressing these requirements

MILS – LET'S SECURE!

Brief MILS History

1975
- Saltzer and Schroeder “The Protection of Information in Computer Systems”

1981
- Rushby “Sesign and Verification of Secure Systems”

1992-2005
- IMA concepts, safety focus

2005
- Hardware virtualisation Intel VT-x, AMD-V

2007
- US Separation Kernel Protection Profile

2007-…
- US-based studies
- The OpenGroup standardisation
- Hardware was not in the focus

2012
- SKPP sun-setted by US government: HW complexity, monolithic eval, politics

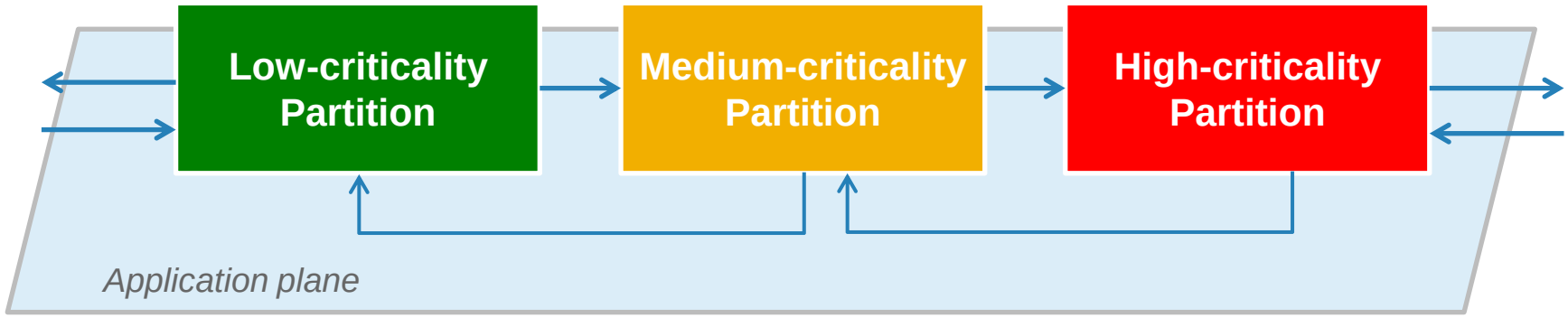
2012
- EURO-MILS
  - MILS Architecture
  - Protection Profile for Separation Kernel
  - High Assurance methods
  - Compositional cert
- D-MILS
  - Research on configuration and analysis of distributed MILS

2014
- Founding of the MILS Community

2016
- CITADEL: adaptive MILS

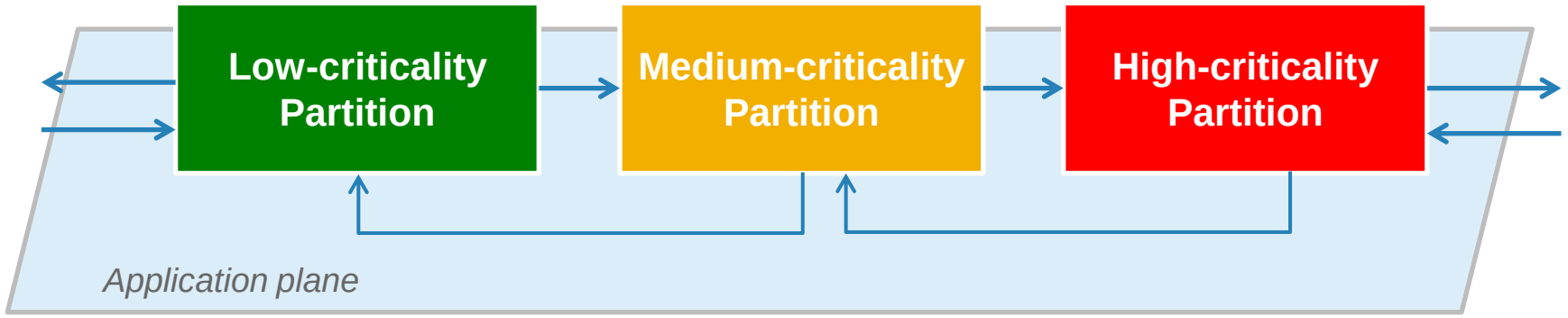
2017
- certMILS: compositional certification of system based on MILS

Developing System Architecture



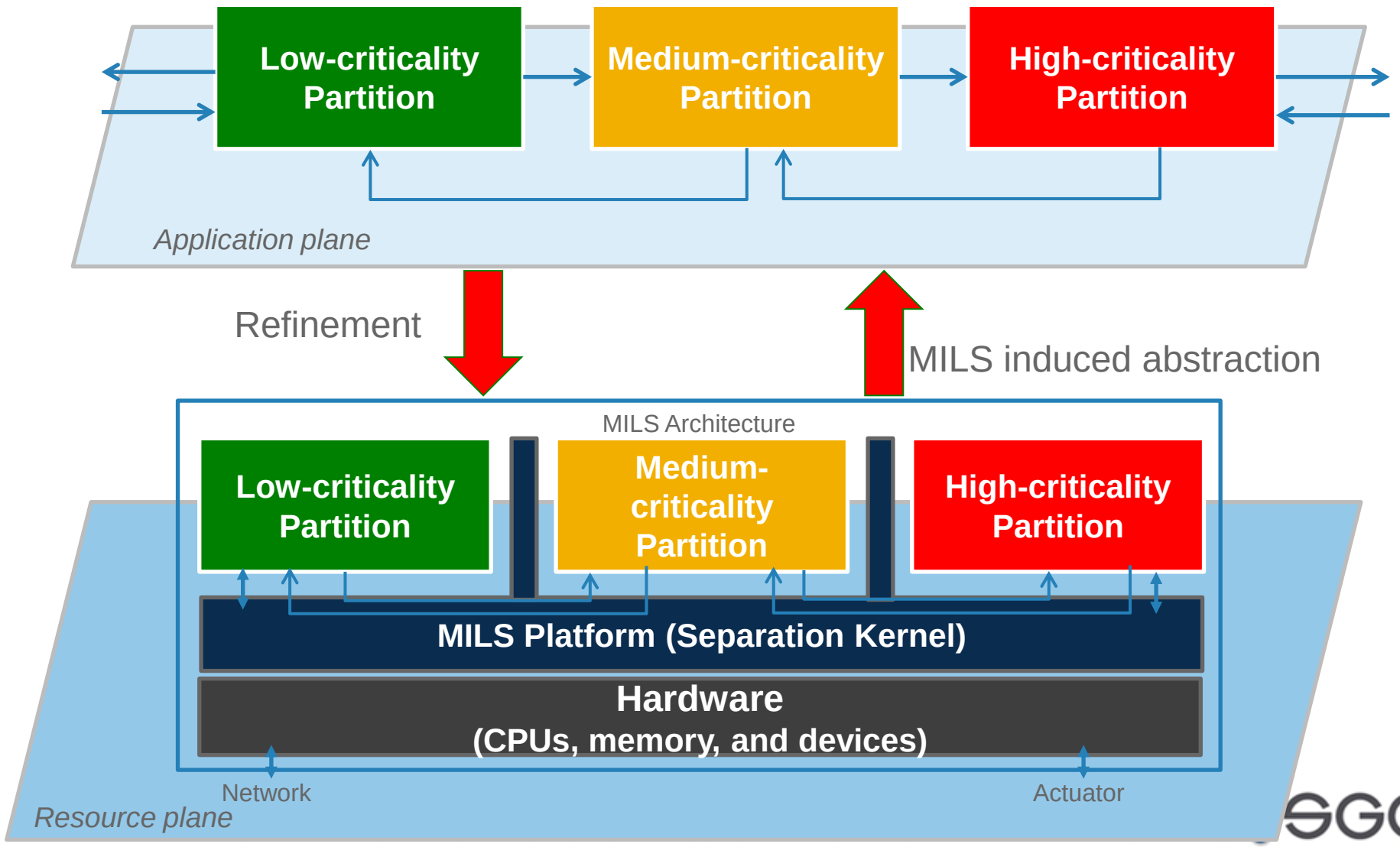
- Generic problems:
  - Composition preserving safety, security, assurance arguments
  - Refinement is a composition, i.e. adding execution environment OS, HW, services
  - Mitigate effects of “have to refine”
    - where we need something to execute systems

MILS



MILS is a **high-assurance security architecture** that supports the **coexistence** of untrusted and trusted components, based on **verifiable separation mechanisms** and controlled information flow

MILS Architectural Approach

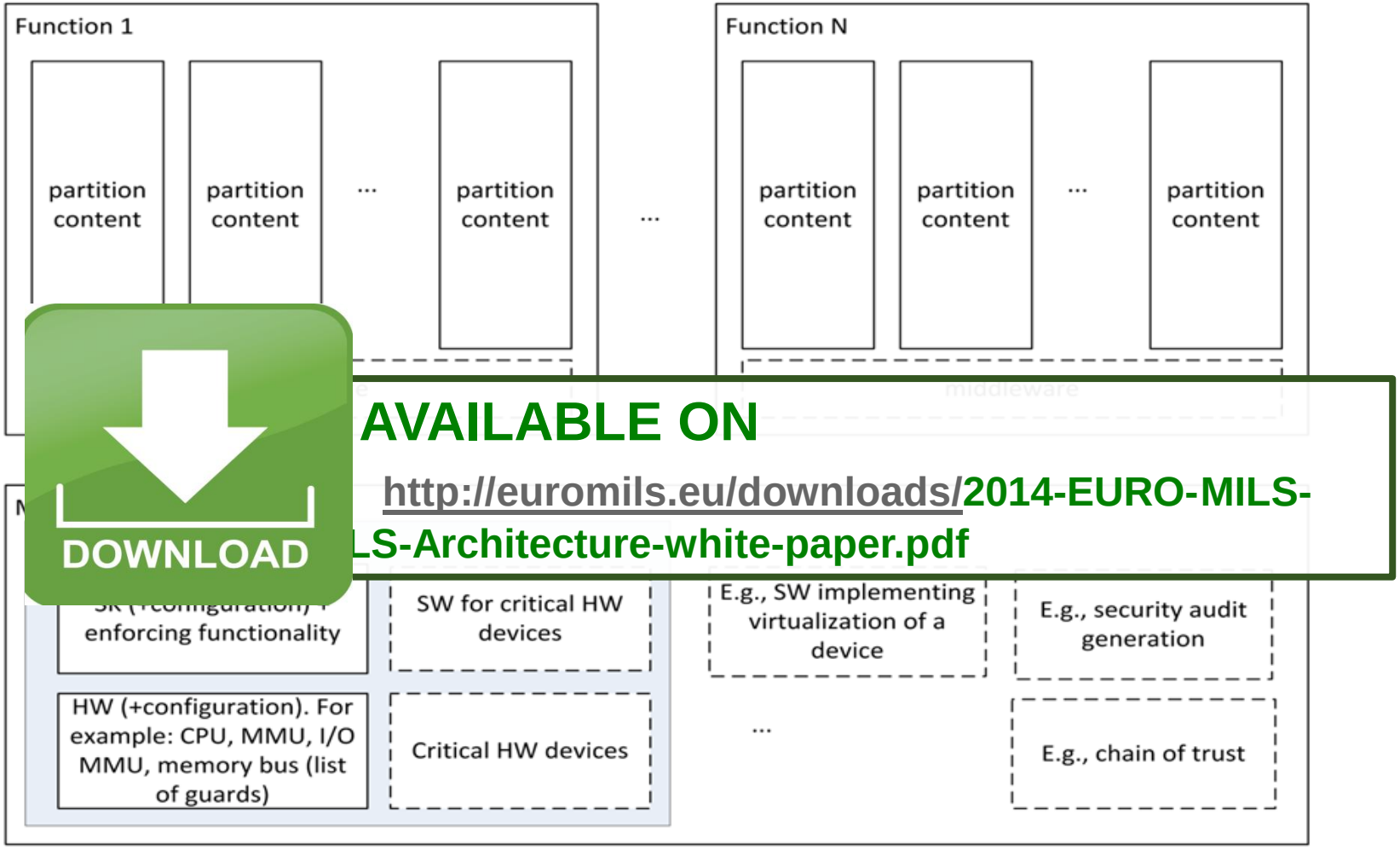


How to build a MILS system

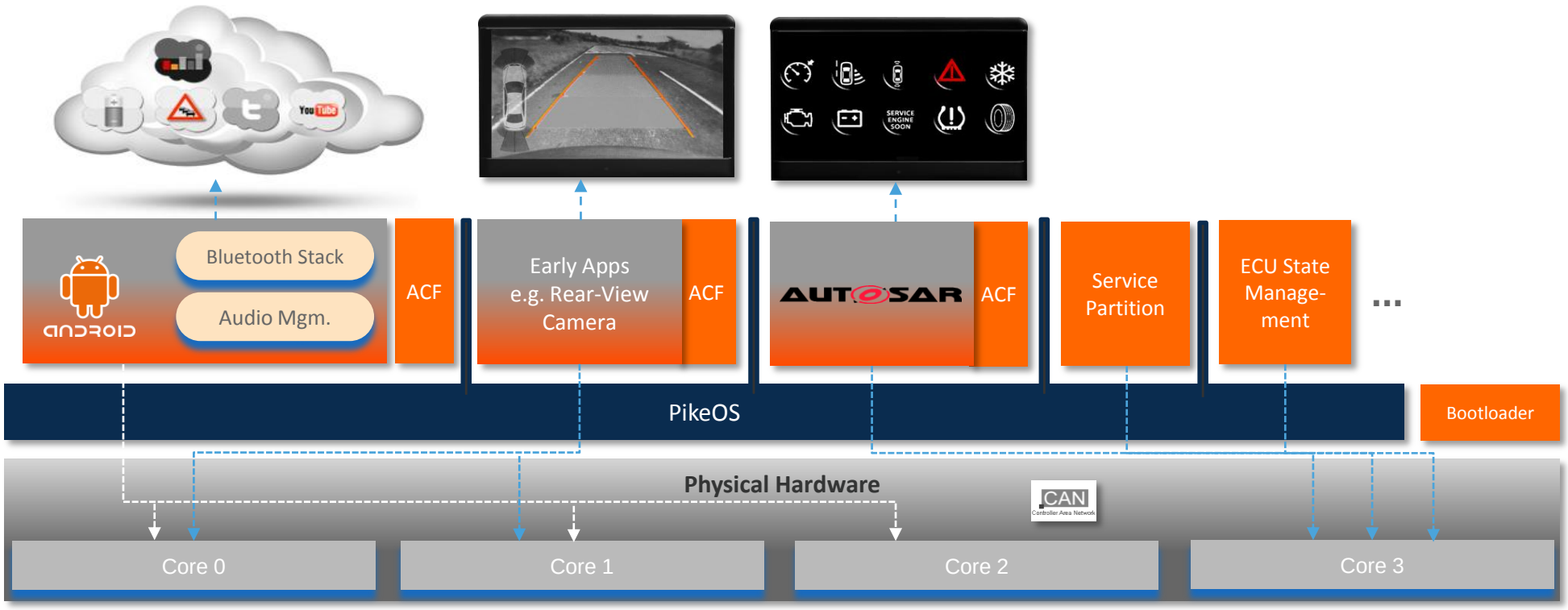
How to use a MILS platform

- **Develop a system architecture consisting of different safety and security domains, i.e. partition system in domains**
- **Assign platform resources to partitions**
  - Assign CPUs, CPU time, memory, I/O devices, file access, available services to partitions
- **Define communication channels between partitions**
  - Default: everything is forbidden what is not explicitly allowed
- **Optionally, add libraries/run-time environments to partitions**
  - e.g. POSIX, ARINC, AUTOSAR, Linux, ANDROID, Ada

MILS Architecture



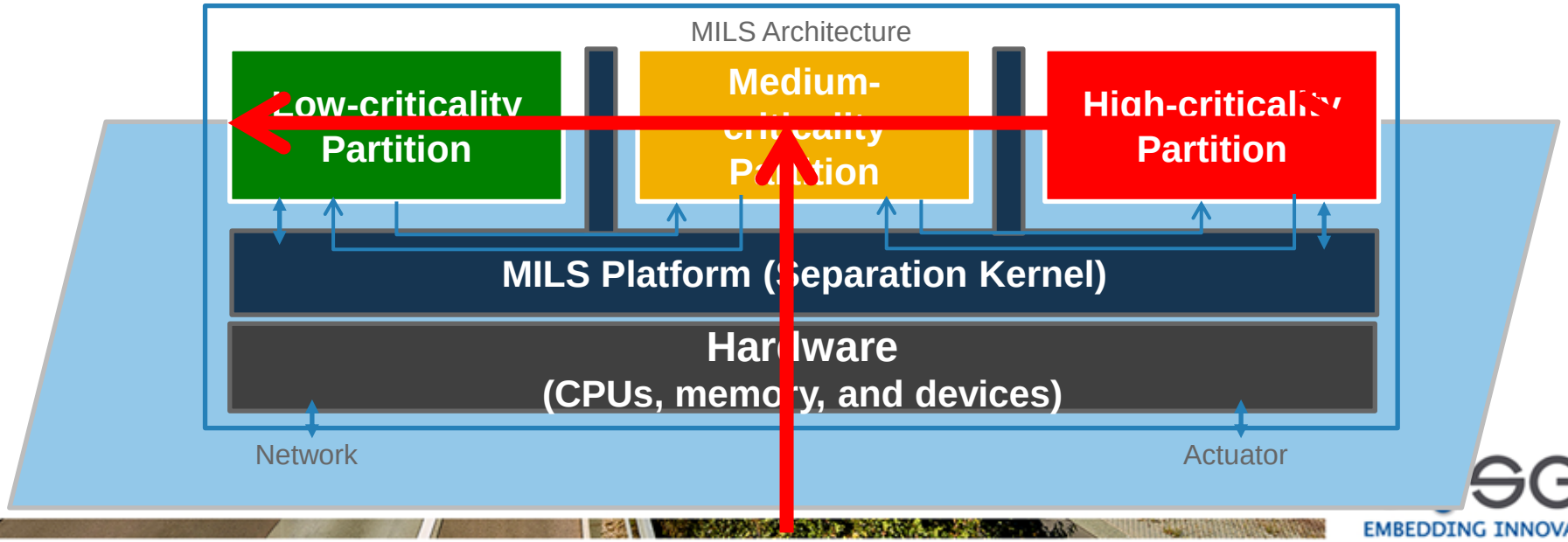
Example MILS in Automotive:
Secure Android-based Head-Unit + Payment Services



Compositional Certification: Scenario-T

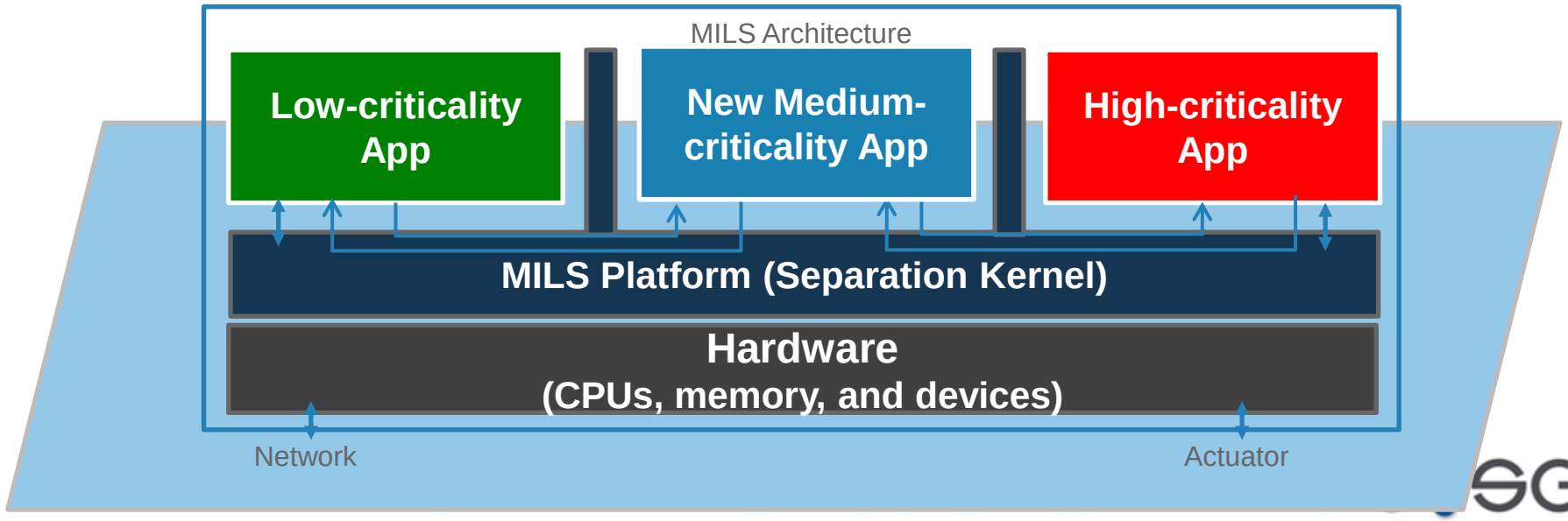
- The core is Separation Kernel
- Components under certified composition
  - Hardware, Separation kernel, Applications

T - composition



Compositional Certification: Scenario-P

- **Puzzle Composition**
  - Exchange system component with interface/function-compatible one
  - Use-cases
    - Product from Vendor-A is replaced by product from Vendor-B
    - Flexible in-the-field update

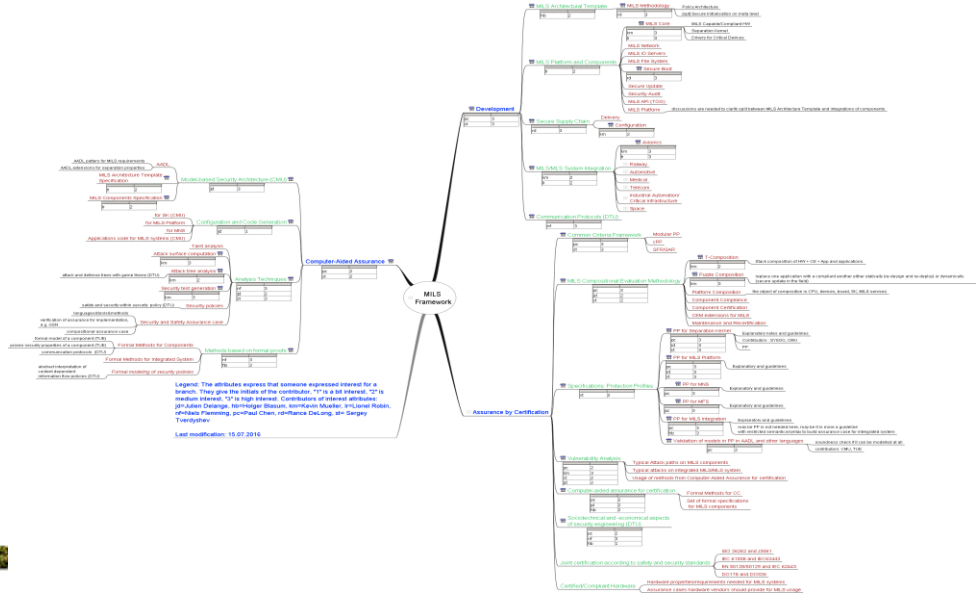


Further Topics/Advances

- **MILS Framework**
- **Certification**
  - Protection Profile for Separation Kernel
  - Compositional Security Evaluation Methodology
    - Baseline for T and P compositions
- **Components**
  - MILS components, e.g. network,
- **Tool based assurance**
  - Formal methods
  - Introspection tools
  - Compositional assurance cases
- **On-going research**
  - Adaptive and Distributed MILS
  - Compositional evaluation

MILS Community: Roadmap

mils-community.euromils.eu



SUMMARY

Summary: Assurance via Certification

	Aerospace	Automotive	Railway
Safety	Long history of standardisation	Recently introduced standard	Long history of standardisation
Security	On-going work on security standard	Staring work on security standard	Staring work on security standard

Security Assurance in Cyber-Physical Systems is **HOT** topic

Takeaways

- Consider adverse actors at the very beginning of the system design stage
- Your system will not be isolated:
  neither physically not information-flow-wise
- System integration concept, i.e. architecture, is
  the most important SECURITY MEASURE
- MILS architectural approach is enabler for
  High-assurance safety and security architecture and
  Compositional certification

SYSGO
EMBEDDING INNOVATIONS

Thank you for your attention!

More information on www.sysgo.com