Applying Kaspersky Security System technology in CITADEL, trustworthy platform for Critical Infrastructure resilience

Ekaterina Rudina
KL ICS CERT
Agenda

1. Project overview
2. Demonstrators
3. Why MILS and what is Adaptive MILS
4. State monitoring based on Kaspersky Security System
5. Challenges and current accomplishments
The CITADEL Project is a collaboration amongst market leading industrial organisations who operate critical infrastructures in Europe, leading software tools and platform technology companies, and research organisations that develop advanced technologies for security and reliability.
What is this project about

• Critical infrastructures are the dynamic systems that demand reliability, robustness, resilience, security, and other attributes.
• These systems while proving high assurance must be developed, certified, deployed, and maintained at an affordable cost.
• To be resilient, a system must be adaptable.

Project implements adaptive MILS in new and evolving adaptive systems contexts having strategic focus within the EU, such as Critical Infrastructures and the Internet of Things, where adaptability is a crucial ingredient for the safety and security of future systems.
Demonstrators

**Frequentis**

**Industrial Demonstration #1:** Frequentis Communication Services. A unique class of communications equipment and software that serves very special purposes in safety of life critical and security sensitive areas (civil and military Air Traffic Control, Emergency Call Dispatching, Police, Ambulance and Firefighters, Coastal and Harbor Control etc.)

**Industrial Demonstration #2:** UniControls / Prague Rail. The objective of the UniControls subway transportation case-study is to develop a novel solution that enhances the security of the existing Prague subway networks.

**Industrial Demonstration #3:** JWO/OAS Manufacturing. The objective of the JWO/OAS manufacturing case study is to demonstrate the use of the CITADEL solutions to enhance security of production facilities, where a control system provider optimises security of the production processes in a manufacturing client’s factory.
Why MILS and what it is about. Assumptions

1. Isolation

These components / connections have no interaction with each other.

C1 → C2

2. Information Flow Control

Only explicitly permitted causality, or interference, is permitted. The architecture permits this flow. Only C1 or C2 can cause the flow, not C3. The flow is directional and intransitive.

C1 → C2
The Roots

Evolution of MILS approach*

The idea behind MILS:
— Secure systems are multicomponent systems that cannot be distinguished from distributed ones
— Separation Kernel is a part of Trusted Computing Base

The goals:
— Safety and Security
— High Assurance
— Support of diverse security policies

1980-2000
Rushby’s work
Separation Kernel as a base for secure systems

2000
Recognition that commercial partitioning kernels for avionic safety applicable to security.
Rediscovery of Rushby’s Separation Kernel

2004
Rushby engaged with MILS community in 2004
Research on MILS funded at SRI International 2004-2012

2008
Separation Kernel Protection Profile.
First commercial implementations

2012
The era of “Modern MILS” 2008-2012 – spawned distributed & dynamic MILS

2016
Distributed MILS, Dynamic MILS, Adaptive MILS, Heterogeneous MILS, Mixed-Critical MILS, Autonomous MILS…

Era of Progressive MILS, 2012-2016 and beyond, built on Modern MILS concepts

*Dates are approximate
Why MILS and what it is about. Policy Architecture

**MILS Policy Architecture**

The architecture expresses an interaction policy among a collection of components.

- **C1**: Trusted Subject
  - Components are assumed to perform the functions specified by the architect (trusted components enforce a local policy).
  - Suitability of the architecture for some purpose presumes that the architect’s assumptions are met in the implementation of the architecture diagram.

- **C2**: Circles represent architectural components (subjects/objects)
  - The absence of an arrow is as significant as the presence of one.

- **C3**: Arrows represent interactions

- **C4**: This component has no interaction with any other

- **C5**: The presence of an arrow is as significant as the absence of one.
Distributed MILS (D-MILS Project)
Adaptive MILS Platform

Application semantics

MILS based system

Classic MILS
Dynamic MILS
Distributed MILS

Adaptive MILS Platform

Autonomous MILS platform

Adaptation and reconfiguration

Reconfiguration
Monitoring

Separation Kernel

Host
Network
Root of Trust

Compositional verification tools

Definition and configuration tools
CII needs to be resilient. The most of CII systems are complex and therefore demonstrate unexpected behavior in case of external impact.

Resilient system is adaptable to external impact.

Some researchers consider adaptable systems as imitating living organisms. **Adaptive MILS is closer to imitation of human behavior.**
The role of State Monitoring (Kaspersky Security System)
Implementation of state monitoring based on Kaspersky Security System
KSS integration with Adaptive MILS platform

KSS Monitor API
- `feed_events(mask)`

Monitoring API
- `part_state(part_id, state)`
- `qport_pending(qport_id, num)`
- `time_part_scheme(scheme_id)`

```
ON QPORT1.PENDING > 3
  TRIGGER EV_QPORT1_PENDING ABOVE 3
ON PART1.STATE = INACTIVE
  TRIGGER EV_PART1_INACTIVE
...  
```

```c
while (true) {
  read_system_state();
  time_part_scheme(...);
  part_state(...);
  ...
}  
```
Examples of informal policies:

- Time between heartbeat events should be no longer than 2 seconds.
- No more than 2 mixers should be running at the same time.
- Time between mixer startups must be no less than 1 second.
- Sensor B value can be greater than 0.8 for no longer than 3 seconds.
- If Sensor D value is greater than or equal to 0.5 then Sensor C can be greater than 1.4 for no longer than 3 seconds.
Project pitfalls

- Varying technology maturity
- Responsibility
- Implementation comprehensiveness
- Integration
- Interaction (14 partners!)
- External control
- Paper work
Questions?