IoT4CPS
Trustworthy IoT for CPS

Novel Design Methods for secure IoT Systems

Achieving IoT Security across CPS Layers

Date: November 2020
Editor: Omar Veledar
Author: Stefan Jaksic

Contents

1. Summary **3**

2. Challenges **5**

2.1. Challenges in IoT-based CPS Systems 5

2.2. Safety & Security Design & Methods 6

3. State of the Art **8**

3.1. Threat Modelling 8

3.2. Building large/scale IoT Systems 8

3.3. Trustworthy Localization 9

3.4. Cryptographic Methods 9

3.5. Security by Resiliency 10

4. Results **11**

4.1. Threat Modelling 11

4.2. Building large-scale IoT Systems 12

4.3. Trustworthy Localization 13

4.4. Cryptographic Methods 14

4.5. Security by Resiliency 15

5. Exploitation Potential **16**

6. Specific Recommendations **18**

References **21**

1. SUMMARY

In order to ensure a high level of security for complex Cyber-Physical Systems (CPS), one must consider security issues already at the design stage. By systematically analyzing potential threats, dependencies and security related requirements at the beginning, a solid foundation for subsequent development, testing and operation is provided.

In this whitepaper we first outline the major design challenges for developing secure CPS. Then we present several results achieved by the IoT4CPS project. In addition, this white paper also advocates that if certain guidelines, processes and recommendations are followed, they increase the likelihood of sustainable deployment of secure IoT systems. Finally, an outlook to potential exploitation of these results is given.

Our security methods and tools tackle challenges along different CPS architecture layers: application layer, platform layer, network layer as well as the lowest level (physical level). Due to space constraints, in this paper we report only on the selected IoT4CPS contributions, highlighted in *Figure 1*.

We start by developing physical-level tools and methods such as sensor security measures for discovering faulty and hacked sensors. We also develop a cryptographic library for forward-secure key exchange mechanism. To develop secure IoT systems and CPS, one needs to understand the design constraints imposed by users and the environment, as well as available technological solutions. For this reason, we report on a recommender system for the development of dependable IoT systems, which helps users to select the appropriate protocols and system configurations for complex CPS. Another method to achieve dependability, applied on a platform-level, is the Self-Healing by Structural Adaptation which allows systems to leverage implicit redundancy to achieve resiliency to failures. Last but not the least, from a security point of view it is crucial to determine the location of IoT devices. This motivated us to research and develop methods for trustworthy localization.

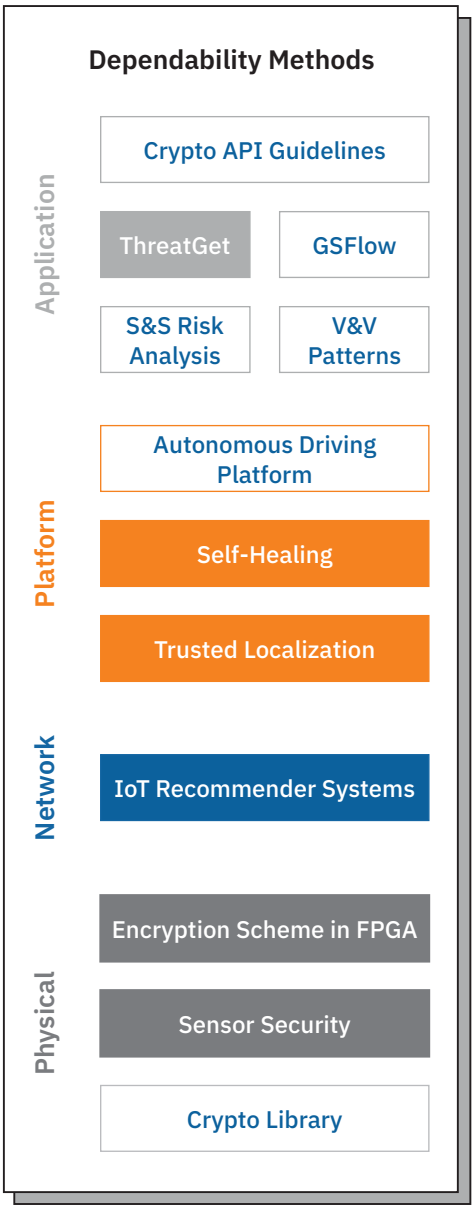

Figure 1: CPS layers and selected IoT4CPS outcomes

On an application level, we report on tools for a variety of tasks in cyber security. We present ThreatGet, a tool that identifies, detects, and understands potential security threats in the foundation level of system models. Moreto is a tool for security requirements analysis and management using modelling languages such as SysML/UML. Our next contribution is a tool for standard-based product development management: GSFlow. It is one of the results of a more general effort to develop tools to support model-based development approaches and Safety & Security by Design. Valuable information on IoT4CPS contributions to other novel IoT and CPS design methods is available in IoT4CPS project website: https://iot4cps.at/deliverables-and-publications/.

2. CHALLENGES

2.1. Challenges in IoT-based CPS Systems

IoT Systems are characterized by increased dynamicity in configuration, system context, system environments and even tasks. Connected systems may change depending on time, con-figuration and goal. The problem of potential emergent behavior, depending on changes and context, which is not easily recognizable during design time and difficult to analyze is a challenge for such systems. In the past, a system could be thoroughly analyzed and tested and potential interactions and goals could be considered during the design process. Updates of system parts were completely under the control of the system operator and usually heavily tested, not done immediately in the live system over the air. With IoT systems, we need to consider uncertainties and work with the unknown. We can no longer rely on covering all risks during design time, we need to continue risk management during the complete system lifecycle.

With the application of IoT in critical environments (production, driving) the safety, security, reliability and resilience of IoT-based CPS systems becomes increasingly important. To ensure these qualities in largely heterogeneous, distributed and dynamic environments, a clear understanding of the properties and how they are ensured during the complete system lifecycle is necessary. The main challenges in IoT-based CPS systems are:

Security – Exchange of critical data: IoT relies on communication and cooperation in order to control, monitor the status and coordinate the system towards a global optimum. The necessary condition for this is trust in other systems and received information. In addition, information can provide deep insights into the behaviour of systems. If this data is accessible by non-authorized parties, privacy and corporate secrets are at stake. Furthermore, unauthorized access to systems and components can raise critical safety threats. This supports the need to validate and protect the data transmitted. Furthermore, access control mechanisms have to be established to ensure that only trusted parties can manipulate involved elements.

Safety – Connectivity of safety critical elements: IoT systems are characterized by an in-creasing level of connectivity. By utilizing IoT elements in safety critical environments, physical safety becomes an issue of the digital domain, as well. This raises the need to consider safety and security together, in order to protect not only the data, but also the physical elements (ma-chines and people) involved. In order to cope with dynamic environments, systems have to be tested even more thoroughly during development, trying to anticipate future configurations and interconnections. During operation, safety properties have to be ensured across sub-systems from multiple vendors, e.g. by fault tolerance features enforcing safe behaviour based on monitoring and the diagnosis of results.

Capabilities – Limited device capabilities: IoT devices are often limited in their computational capabilities due to power and resource limitations. Thus, highly efficient security mechanisms with a sufficiently small footprint are needed in order to ensure safety and security on these devices.

Diversity – Increasing number of participants involved: Industrial IoT systems involve hundreds or thousands of different elements from different manufacturers. In order to ensure safe-ty & security of such systems, all elements need to be trustworthy and reliable. This requires a common understanding of trust and mechanisms for trust validation and management.

Acceleration – Decreasing time from design to production: One of the main challenges for solution providers of autonomous driving functions and components is to offer an appropriate environment which enables the efficient development, validation, instrumentation and finally deployment of innovative solutions. To ensure safety & security in these highly accelerated processes, methods and tools for constant validation and monitoring of the stated requirements are needed.

Maintainability – Maintenance over whole life cycle: Industrial products have a typical time of operation between 10 and 20 years. In order to ensure safety & security over the whole life cycle of such systems, constant monitoring and adaptation to new requirements and threats is necessary.

None of these topics are completely new, but in IoT-based systems they advance to a new level and need to be addressed in combination.

2.2. Safety & Security Design & Methods

A secure system can be designed and developed only if security issues are well-identified and addressed appropriately in the early stages of the system development. That is considered a significant advantage because once the system is developed, the introduction of security countermeasures becomes unfavorable. **Threat modelling** is a structured process of examining a system for potential weaknesses, and in the second phase, the process of resolving those. It is a systematic approach based on a conceptual model of weaknesses and threats. Through a series of iterative updates, the process constantly improves the weaknesses and threat model. Threat modelling can be applied throughout the product lifecycle: during the conceptual phase, during development as well as during the operation (shown in *Figure 2*).

The concepts described in IoT4CPS may be used to protect stakeholders from wrong integration of automation devices in the production environment and from attacks on the setup process. The main security challenge in the field of **localization** that is related to the use-case in question is to prove that a device is at a specific location. We initially consider a trusted scenario, in which the third-party employee is honest and the environment is free of attacks. The goal of trustworthy localization is to prove to the operator that the device was correctly installed in the specified location within its production environment.

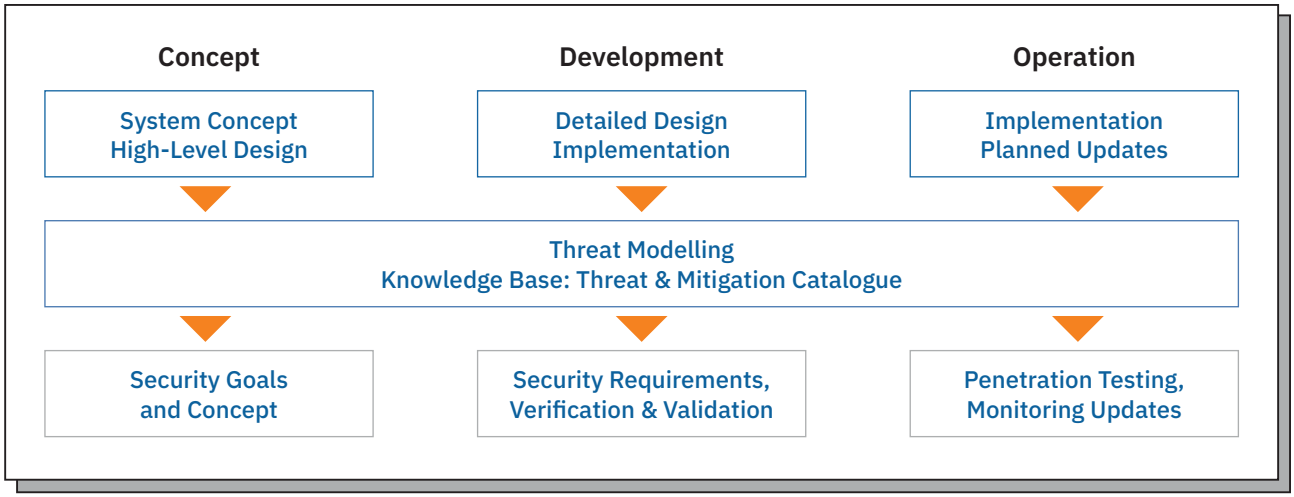

Figure 2: Threat Modelling outputs throughout the lifecycle

Building trustworthy IoT applications relies on the capability of the IoT device to securely communicate with each other or also with central services. The main challenge in achieving such **secure communication** in IoT application is their distributed nature as well as the limited computing resources of IoT devices. For the software side, our focus is on the design of the APIs of cryptographic libraries to make them easily accessible to application developers, but also to ensure their correct usage. On the hardware side, we bring guidelines for the secure implementation of lightweight symmetric encryption algorithms on Field Programmable Gate Arrays (FPGAs), so that protocols for secure communication can also be deployed on resource-constrained devices and legacy hardware.

3. STATE OF THE ART

To correctly understand and evaluate the contributions brought by IoT4CPS project it is important to present the state-of-the-art in the relevant topics: threat modelling, IoT system implementation constraints, trustworthy localization methods, cryptographic key-exchange methods and cryptographic API usability, as well as security by resilience.

3.1. Threat Modelling

Threat Modelling is an important method for achieving system security [MLY2005]. The Threat Agent Risk Assessment (TARA) [ROS2009] and STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privileges) methods are considered as the most suitable threat modelling frameworks in the automotive industry. The TARA method is based on three main threats libraries which is used to define the most common methods to attack assets, defining the most exposed areas. The STRIDE method gives the flexibility to identify potential threats in the design phase of any software and hardware applications [KKA2017]. These threat modelling methods are applied to the automotive and railway domain, as discussed in [KKA2017] and [MSCH2016], [STHS2019], respectively.

Cyber-security risks and attacks on connected devices are well-documented [VMAJ2019], [ZG2013]. Novel methods for security threat analysis include using ontology-based models, as proposed in [SSG2019]. Threat modelling is an integral part of automotive security approaches for identifying potential threats and specifying corresponding security mitigation [EVI2009], [HEA2016]. One introduced threat modelling method is based on existing tools for different phases of the development life cycle, such as concept phase, product development, production and operation [MSCH2016]. It is important to note that threat modelling integrates well into the development lifecycle based on security standards [SKS2018] such as IEC 62443 [ISA2018] standard for industrial security.

3.2. Building large-scale IoT Systems

Due to the continuous expansion of IoT technologies, the architects of such systems are facing challenge when it comes to satisfying different constraints to build cost-effective and efficient IoT systems which satisfy the functional requirements. Today, developers have a huge set of connectivity networks at their disposal, ranging from short-range networks such as Bluetooth to global connectivity via satellite networks [VVC2018, ERI2016]. Depending on the target application, the system architect must decide between one or another set of IoT technologies. A 2-step methodology which would guide the IoT system architects has been proposed in [VVC2018]. Several survey papers highlight the pros and cons of applying different state-of-the-art technologies to build large-scale IoT systems [ESC2016, SC2016].

3.3. Trustworthy Localization

A solution using existing WiFi Access Points (APs) is described in [SW09] in which a location-proof protocol is established using the proximity of a client to a set of trusted APs with a known location. Four main disadvantages are listed in this paper: (1) the orientation of the anchor is unknown, (2) the first localization of a tag is established by guessing its position (3) a high number of ranging messages is needed and, (4) all antennas within a localization estimation must be within communication range. The achievable accuracy of WiFi-based approaches depends on many factors, such as environmental changes, AP density, unstable Received Signal Strength Indicator (RSSI) measurements and Line-of-Sight (LoS)/Non-Line-of-Sight (NLoS) conditions. In practice, one can expect with this technology a mean error above 4m.

Another solution entitled “LINK” [TCB15] uses short-range wireless connected devices such as Bluetooth transceivers and validates location claims based on a centralized analysis of spatio-temporal correlation between the users, trust scores associated with each user and historical trends of the trust scores.
In [BLFC15], a framework compatible with (and relying on) short-range technologies, such as BLE and iBeacon, is proposed. However, the granularity of the proof may still not be enough for the use-case in question, since it aims at room-level accuracy only (typical mean error between 1.6m and 2.5m).

Given the strict accuracy requirements imposed by the use-case and the inapplicability of GPS due to accuracy and scenario, which may be indoors, Ultra-wideband (UWB) transceivers can be used. They are able to provide sub-decimeter accuracy and are resilient to multipath. Prior work proposed a UWB-based system called SALMA capable of localizing a tag with a single-anchor [GRKB18] assisted by multipath components. Nonetheless, SALMA requires the knowledge of the surroundings of the environment where the localization takes place, which is undesirable in many practical use-cases.

3.4. Cryptographic Methods

One of the major protocols that provide a secure communication channel such that both authenticity and confidentiality of the transferred data can be ensured is Transport Layer Security (TLS) [RES2018]. However, due to the complexity of its state machine and the usage of public-key infrastructure, it is not the number one choice for IoT devices. It also offers a large variety of choices for the selection of cryptographic primitives [CSF2008]. Yet, the security properties guaranteed by TLS are of paramount importance for building a trustworthy infrastructure.

One central ingredient to secure today’s Internet are key exchange (KE) protocols with the most prominent and widely deployed instantiations thereof in the Transport Layer Security (TLS) protocol [RES2018]. Using a KE protocol, two parties (e.g., a server and a client) can establish a shared secret (session key) which afterwards can be used to cryptographically protect data to be exchanged between those parties. The process of arriving at a shared secret requires the exchange of messages between client and server, which adds latency overhead to the protocol. The time required to establish a key is usually measured in round-trip times (RTTs). A novel design goal, which was introduced by Google’s QUIC protocol [TI2017]

and is also adopted in TLS version 1.3 [RES2018], aims at developing zero round-trip time (0-RTT) protocols with strong security guarantees. So far, quite some effort was made in the cryptographic literature, e.g. [WTS2016, HJL2017], and, indeed, 0-RTT protocols are probably going to be used heavily in the future Internet as TLS version 1.3 adoption is growing rapidly. Besides TLS 1.3, Google's QUIC protocol is used on Google webservers and within the Chrome and Opera browsers to support 0-RTT. Unfortunately, none of the above-mentioned protocols are enjoying 0-RTT and full forward secrecy at the same time. Only recently, Gunther, Hale, Jager, and Lauer [GHJ2017] made progress and proposed the first 0-RTT key exchange protocol with full forward secrecy for all transmitted payload messages. However, although their 0-RTT protocol offers the desired features, their construction is not yet practical.

3.5. Security by Resiliency

Many architectures and approaches for runtime adaptation are based on the service-oriented architecture (SOA) [MLM2006], [OMG2012], where a specific functionality of the system is encapsulated in a service. Our demonstrator uses a middleware that is based on the SOA approach to facilitate the reconfiguration mechanism. However, SOA is a very general architecture that does not deal with properties like embedded, real-time or safety-critical - properties related to CPSs. [CHE2009] surveys the state-of-the art and research challenges in adaptive systems. [DHT2002] presents technologies and methods enabling SHSA using xADL that is a language to describe the architecture of a system. The authors in [SWG2014] give a thorough introduction and review of the trends and research of large-scale loosely coupled CPSs and discuss general key requirements and challenges of an ”elastic” or a dynamic CPS.

Data Distribution Service (DDS) [SCH2005] is a communication infrastructure (similar to ROS publish/subscribe) for heterogeneous CPS. It provides Quality-of-Service (QoS) policies which may be used to monitor and ensure real-time properties (e.g., bounds for message latency or jitter). MQ Telemetry Transport (MQTT) is a popular communication protocol for IoT applications also based on publish/subscribe. Compared to DDS it is lightweight, but due to its centralized architecture (all data is sent to a broker) more suitable for sporadic data exchanged. See [FGM2015] for an overview about protocols and standards for IoT. Related work to ORR can be found in [HO2013]. Compared to ORR, there are other techniques that may modify the behavior of components themselves, rather than the system's structure. This is often referred to as software adaptation [OMT2008] or parametric adaptation [CHE2009]. This kind of adaptation mostly depends on the abilities of the components themselves and their interfaces, thus not in focus of our work.

4. RESULTS

In this section we present the solutions to the selected challenges, brought by the IoT4CPS project.

4.1. Threat Modelling

To respond to the challenges posed by the need for the cyber security of resource-restricted devices, IoT4CPS shines the light on practical engineering method and building blocks for dependable IoT systems. Although this white paper is centred around our security tools such as ThreatGet, we highlight other notable contributions. These methods and tools for building dependable systems include a cryptographic library for a light-weight key-exchange mechanism, guidelines for developing usable cryptographic APIs, recommender system, trustworthy localization as well as methods for system resilience.

The threat modelling is an iterative process comprised of several typical steps. The starting point is the system modelling with all security assumptions. In the next step, the potential adversaries and threats are added. Then, the threat model is applied to the system model to identify potential threats. The ThreatGet evaluates all identified threats and decides on the risk treatment. The system model is updated with security measures. If there are some unidentified threats, the threat model is updated. Upon such an update, it is necessary to repeat the evaluation and system model update step. Once there are no new threats to introduce, one last iteration of this process performs full threat evaluation with security countermeasures.

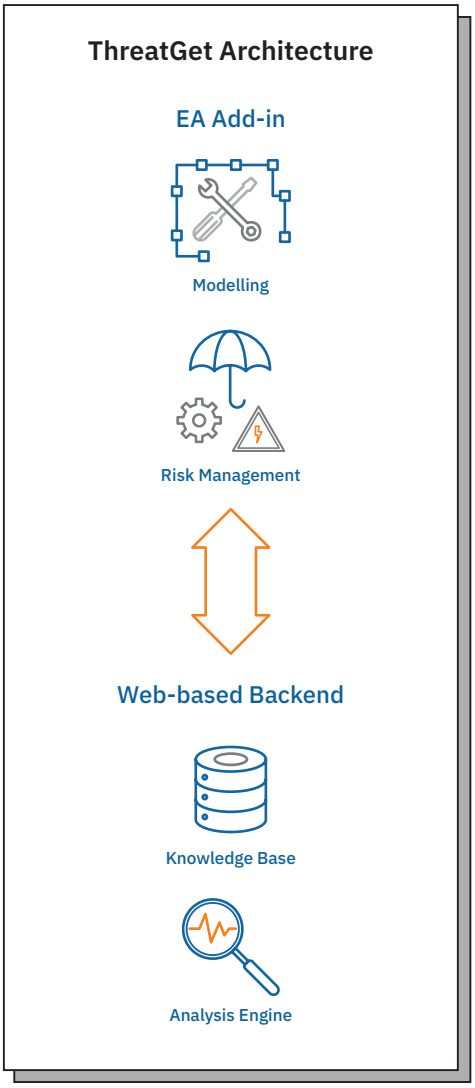

Figure 3: The iterative process of several typical steps that make up the threat modelling.

Figure 4: IoT-based Smart Factory illustrates an example of IoT application in a smart factory, which is one of two main use cases of IoT4CPS project. The example contains one or more sensors and camera units for gathering information about the production line. The collected data is processed by a control unit that handles and manages actions by sending signals to actuator units such as robotic arm and engine. The data is sent to a centralized data storage and processing unit for monitoring the production quality.
In such a heterogeneous and distributed system, potential security threats can emerge in any component, which may compromise the operation of the entire system. To identify potential threats in the system, we apply ThreatGet. The threats are defined according to the dataflow from the source components to the targets. According to the security properties of these units, some vulnerabilities could be exploited.

In *Figure 4*, the Control Unit takes the central role in the Smart Factory model. This component communicates with the data storage through a gateway, which runs a certain communication protocol. Depending on the gateway device, it can provide low or high security features. In our model, we can analyze devices with different levels of security by adjusting the possible security parameters of the model. We model possible security mitigation measures in the communication flow between the Control Unit and the Gateway. The features include: Source and Destination Authentication, Confidentiality and Integrity. ThreatGet detects a number of potential threats, which are classified according to the STRIDE model (i.e., Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service (DoS), and Elevation of Privilege).

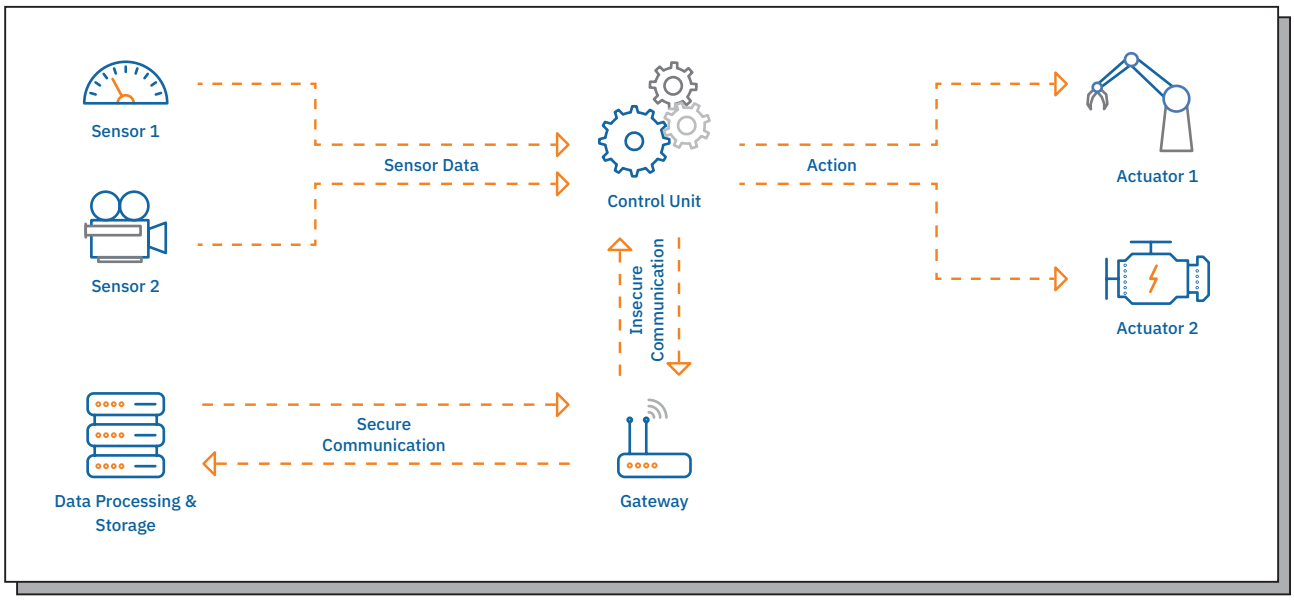

Figure 4: IoT-based Smart Factory model in ThreatGet

Spoofing represents an attempt by a person or program to identify itself as another by falsifying data, to gain an illegitimate advantage. Data tampering is an attempt to maliciously modify the data through unauthorized channels. Repudiation is a kind of attack which manipulates the log data in the computer systems, to conceal traces in the log. Denial of Service and Elevation of privilege are well-studied threats, where an attacker is jamming the access to the system resource, and when an attacker attempts to gain more access rights than allowed, respectively.

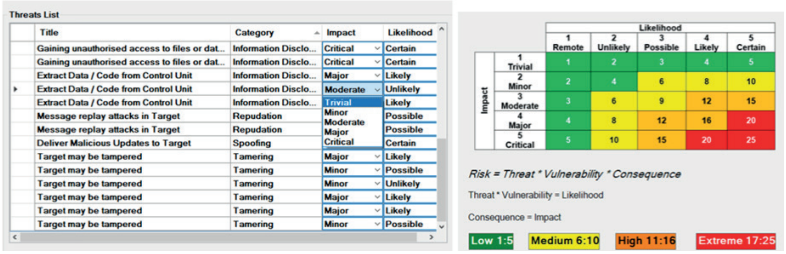

Figure 5: Identified Threats and Risk Assessment Chart as a result of ThreatGet

4.2. Building Large-Scale IoT Systems

Besides security tools, IoT4CPS brings several other notable results, which contribute to dependability methods. Along with the growing market of Industrial IoT (IIoT) applications, the set of available network technologies is continuously expanding. Today, developers have a huge set of connectivity networks at their disposal, ranging from short-range networks such as Bluetooth to global connectivity via satellite networks. Depending on the specific use case of each IIoT application, different approaches constitute the most cost-effective network technology solution, as there is no “one-size-fits-all” solution.

Complex IIoT systems have a significantly larger set of dependability requirements compared to “normal” IoT applications. As the systems connectivity network plays a major role in fulfilling these requirements, choosing the correct set of technologies is crucial. To build a reliable and functional IoT ecosystem, one can use a **recommender system**, developed by Siemens. The system designer has to provide application-specific requirements (e.g., purpose, location, connectivity, power supply) and high-level architecture patterns (e.g., direct connection field cloud or multilevel communication via edge devices). Based on this information, the recommender system can compute a feasible system setup (system topology, technologies to apply within the system) without the need of consulting experts.

4.3. Trustworthy Localization

A wide range of system-critical applications in various domains such as autonomous driving or industry 4.0 actively utilize localization. To apply localization mechanisms in the context of these applications, it is important to assure their secure and dependable performance. Our partners of TU Graz evaluated the existing approaches and proposed new methods for **trustworthy localization**. When we talk about localization we focus on sensor fusion. Many different filter algorithms can be used for sensor fusion in navigation systems. One of the most popular approaches is the extended Kalman filter (EKF), which simultaneously estimates the sensors systematic errors and corrects the positioning errors. However, extended Kalman filters inherently assume that both the process (system) errors and measurement noise (observation errors) are Gaussian distributed. Applying extended Kalman filters in case of non-Gaussian noise is not straightforward.

Therefore, in IoT4CPS project we proposed a stochastic approach based on particle filters (see *Figure 6*) for fusing the Ultra Wide Band (UWB) and Inertial Measurement Unit (IMU) position data, as such an approximate approach could tackle the multimodal distribution of errors without normality assumptions. Particle filtering uses a set of particles (samples) to represent the posterior distribution of a stochastic process given noisy and/or partial observations. It is a well-established methodology for generating samples from distribution without making assumptions about the state-space model or the state distributions. The state-space model can be nonlinear and the initial state and noise distributions can take any form. Particle filters implement the prediction-updating step in an approximate manner. The samples from the distribution are represented by a set of particles; each particle has a likelihood weight assigned to it that represents the probability of that particle being sampled from the probability density function. In order to prevent a weight

collapse when the weights become too uneven, a resampling step is used in which the particles with negligible weights are replaced by new particles in the proximity of the particles with higher weights. As long as there are sufficient particles available a reliable approximate estimate can be obtained efficiently.

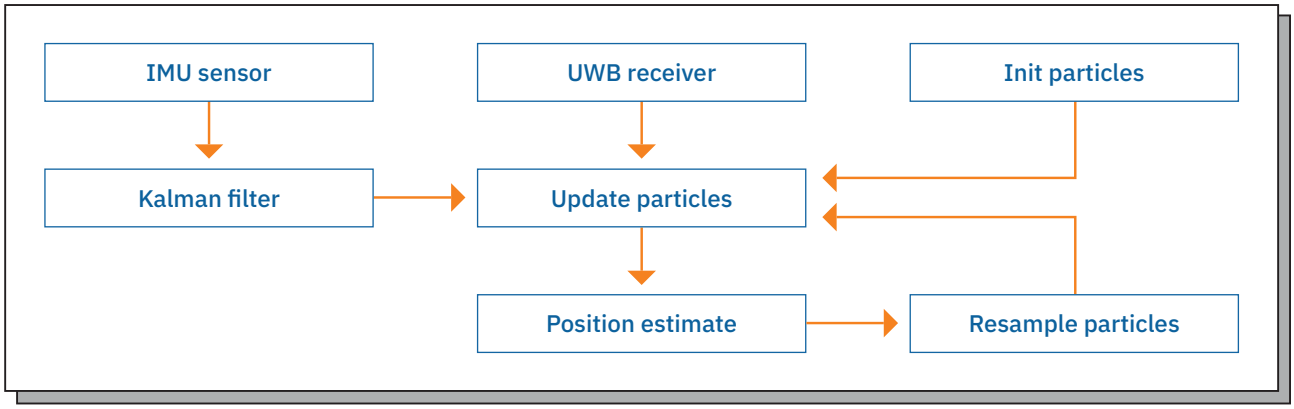

Figure 6: Schematic diagram describing IMU/UWB data fusion using a particle filter

The proposed probabilistic approach integrates two different technologies for position tracking and as such offers reliable fall-back options in case of intermittent loss of operability of one of the technologies. Such situations can arise due to environmental constraints; however, they could also be triggered by an indirect attack targeting the accuracy, the integrity and the continuous flow of sensor data. To mitigate the potential effect on the tracking dependability of a multitude of uncertainty sources, some of which are unpredictable and previously unseen, we tackle the uncertainty with a probabilistic framework providing approximate, but reliable position estimates as well as the associated confidence levels. This complements the methods dealing with more direct forms of cyber-attacks, specialized on compromising the digital content by tampering, spoofing or repudiation. Also, this approach could offer higher resilience to attacks targeting the accuracy, the integrity, or the continuous flow of sensor data used in the position and orientation tracking. Particle filters could mitigate the detrimental effect of uncertainty created by such attacks by providing an approximate estimate and a confidence range based on the particles' distribution. Integrating two different technologies for position tracking (UWB and IMU) within this probabilistic approach will allow for a smooth and reliable fall-back operation in cases when the system's integrity is compromised.

4.4. Cryptographic Methods

We describe our results in novel cryptographic methods, developed in the context of IoT4CPS. In consent with the concrete project and WP objectives regarding low-latency and scalable cryptography (including identity-based techniques), this results in:

- A more efficient low-latency and forward-secure key-exchange scheme [DGJ2020].
- A low-latency and forward-secure identity-based scheme for a more scalable key-exchange scheme with fine-grained access control [DRS2020].
- A post-quantum (i.e., long-term) secure low-latency key-exchange scheme from generic identity-based cryptography techniques [CRS2020].

For devices that are more powerful and capable of running a typical TLS stack in software, we focus on a new property of the protocol that was introduced in TLS 1.3: zero round trip time (0-RTT) key exchange. This feature enables TLS clients to immediately start sending encrypted data to the TLS server without completing the full handshake first. Thereby, the additional latency introduced by the handshake of TLS can be significantly reduced. However, in the current version of the protocol, 0-RTT key exchange is only possible after at least one successful connection between the client and the server and additionally requires them to store some state in the form of a shared secret. In an IoT setting, where we have servers powerful enough to enable TLS and potentially weak TLS clients, puncturable key encapsulation schemes facilitate the implementation of 0-RTT key exchange without storing shared secret data on the devices. For the successful integration of this technique in TLS, we provide an implementation of a puncturable encryption scheme as library. Thus, we also derive **guidelines** for the implementers of cryptographic schemes to ensure that they can safely be used by software developers and integrators.

In IoT4CPS project we focus on the lower-level cryptographic primitives that are employed in TLS and the surrounding infrastructure. We derive guidelines for their secure implementation in hardware, especially on FPGAs. In particular, we focus on lightweight primitives including lightweight **authenticated encryption schemes**. As a result, we developed the authenticated encryption scheme ISAP for FPGAs and ASIC which is especially useful for constraint devices. The ISAP allows on the one hand for fast software implementations, but more importantly in the context of IoT4CPS, also for fast and compact hardware implementations. Additionally, ISAP is designed to be resistant against passive side-channel attacks such as timing attacks, differential power analysis, and against active attacks including fault attacks.

The complexity, insecure defaults, and poor documentation of cryptographic APIs are among reasons for frequent developer-induced errors in applications that subsequently lead to security incidents, such as numerous applications validating TLS certificates incorrectly. For this reason, SBA research contributed to our overall cryptographic results with their guidelines for developing **usable cryptographic APIs**. These guidelines, presented in IoT4CPS deliverable D3.5, explain to developers how to architect crypto APIs that minimize chances of being misused.

4.5. Security by Resiliency

TU Wien and TTTech contribute with a method for towards improving system resilience called **Self-Healing by Structural Adaptation (SHSA)**. Self-healing is the ability of the system to react also to failures not specifically considered during design-time, e.g., faults caused by functional, environmental or technological changes or zero-day malware. A very promising approach of achieving self-healing is through structural adaptation, by replacing a failed component with a substitute component by exploiting implicit redundancy. The SHSA algorithm uses a knowledge base, modeled as an ontology which defines the interrelation of properties in the CPS as well as additional runtime information of the CPS with the goal to discover the implicit redundancy in the system.

5. POSSIBLE EXPLOITATION

IoT4CPS identifies possible usage of the ThreatGet tool [IoT4CPS deliverable D3.2] to create a threat model of an IoT-based smart factory in order to increase overall system security. ThreatGet enables the creation of reusable and extendable threat models that can be implemented in several generations of a product as well as in different lifecycle stages. ThreatGet helps in **lowering development** cost by avoiding re-spinning of the development process. The tool is also applicable in any application domain which requires security analysis. When we speak about the domain of Automated Driving, we identify **Advanced Driver Assistance Systems** and respective hardware platforms for their development as an interesting use case. During the development of these systems, safety is of utmost importance. However, safety solutions can be trusted only if a full consideration and proper treatments are given to security issues.

ThreatGet is evolving with new features and improvements. Its internal knowledge database is maintained and constantly updated new threats. To define a detailed threat model, one needs to take into consideration assets, potential threats, vulnerabilities, exploits and security requirements in a holistic view. Management and ordering of such large data is cumbersome. Hence, ThreatGet is evolving towards an **ontology-based approach** to define relationships between these entities. Such an approach will allow for manipulating with an enormous amount of vehicle data to ensure the highest level of security i.e. for modern vehicles. The current commercial version is offered by LieberLieber GmbH and AIT Austrian Institute of Technology and is aimed at checking vehicle cyber security aspect. Its success proves the conversion of the exploitation potential into industrial usage.

ThreatGet is complemented with other tools i.e. MORETO and GSFlow, which enhance the management of the whole product development concerning the safety standards (complete life-cycle). MORETO improves handling of security requirements analysis, allocation, and management using modelling languages such as SysML/UML. Just as the ThreatGet, it is an Enterprise Architect plugin for managing the IEC 62443 security standard. On the other hand, GSFlow improves the management of the product life cycle. Together, these tools comprise a toolchain that handles a range of different aspects of the security analysis. The toolchain will be further enhanced towards expanding its scope and maturity needed for industrial applications.

Two major achievements in terms of cryptographic methods for dependable IoT systems are the implementation of forward-secure zero round-trip time (0-RTT) key exchange protocols based on bloom filter encryption and authenticated encryption scheme ISAP for FPGAs and ASIC, useful for constrained devices. We provide a library which is also integrated as part of the widely-used TLS library OpenSSL. Thereby, we obtain an **extension of the TLS protocol** that allows resource constraint devices operating as clients to send data via a secure channel to a server without waiting for server to reply.

Therefore, for the case of devices with very limited resources, which IoT devices certainly are, our authenticated encryption scheme ISAP will enable to deploy **new algorithms on legacy hardware** without disrupting their runtime guarantees. This will also improve trustworthiness of IoT applications that rely on resource-restricted hardware.

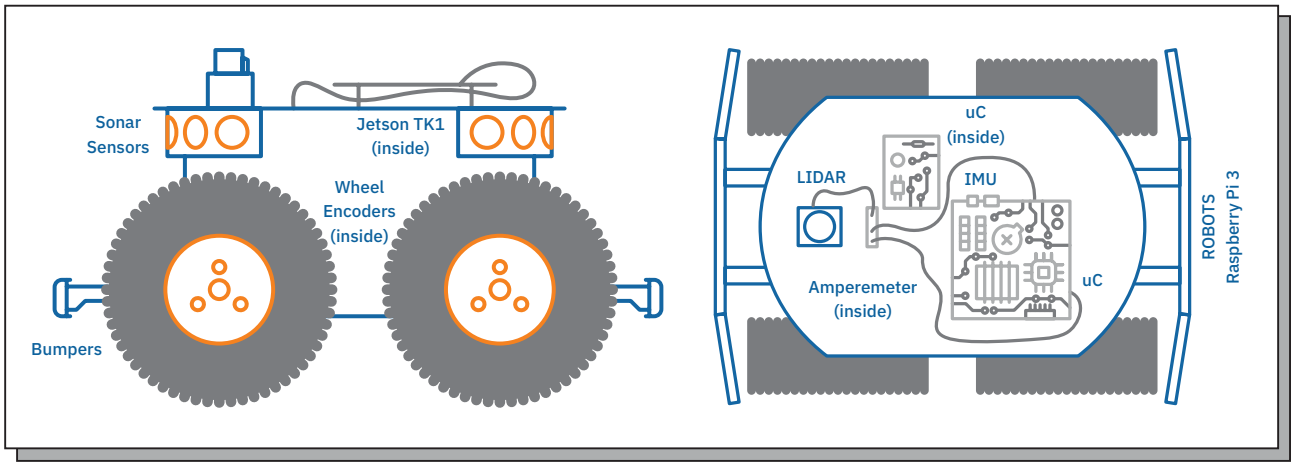

Figure 7: Mobile robot equipped with its sensors and processing units

Self-Healing by Structural Adaptation was implemented on a mobile robot (*Figure 7*). IoT4CPS [deliverable D6.1] shows how to deploy this resilience algorithm on a mixed-criticality platform, which is a common platform for developing **automotive** features. This way we demonstrate how SHSA can be integrated into the automotive architecture to allow for greater system resilience.

Further details on all the project results can be found in IoT4CPS deliverables and publications, available at: https://iot4cps.at/deliverables-and-publications/.
For a practical intro on ThreatGet please refer to threatget.com.

6. SPECIFIC RECOMMENDATIONS

Dependable IoT solutions are improved with holistic concepts, which is to be complemented by an appropriate toolchain. While each focusing on its own scope, the tools should complement each other, hence, capturing the complete spectrum of challenges. **Their advantage is a flexible methodology, which is independent of the application domain.** There is an expanding range of such domains, which require solutions, hence offering a chance for increased exploitation in industrial domains.

Providing that it is applied to a hardware platform, IoT4CPS tools are directly portable to other domains. Should successful exploitation occur, the portability is also a highly recommended aspect for any development within the IoT world. **The dependability methods and tools ought to be developed in a generic and domain agnostic manner.** They should come into consideration at the design stage, as adding dependability methods in post-design stages is a cumbersome and complex process with the potential to expose opportunities for cyber attacks due to a non-holistic solution to the challenge.

Designing a complex IoT ecosystem in a reliable and functional manner could benefit from the usage of a computational recommender system, which relies upon analysis of specific requirements and high-level architecture patterns.

Trustworthy localization is a necessity in many IoT applications. This goes beyond security and reaches deep into the safety aspect of the many domains and their applications. One must pay attention evolution of the available techniques not to be left behind in terms of development. This kind of evolution is also followed by the evolving data encryption methods, which are crucial for secure data transmission and avoidance of security incidents.

A crucial aspect of all proposed design techniques, methods and tools is intuitive usability. An increasingly promising method of ensuring system resilience is seen in self-healing techniques. These offer a proper adaptation to failures and attacks.

REFERENCES

[BLFC15] Jacob T. Biehl, Adam J. Lee, Gerry Filby, and Matthew Cooper. 2015. You're where? prove it!: towards trusted indoor location estimation of mobile devices. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp ,15). ACM, New York, NY, USA, 909–919. DOI: https://doi.org/10.1145/2750858.2804284

[CHE2009] B. H. C. Cheng et al., "Software engineering for self-adaptive systems: A research roadmap," in Software Engineering for Self-Adaptive Systems. Berlin, Heidelberg: Springer Verlag, 2009, pp. 1–26.

[CRS2020] V. Cini, S. Ramacher, D. Slamanig, C. Striecks. "CCA-Secure (Puncturable) KEMs from Encryption With Non-Negligible Decryption Errors." In submission, 2020.

[CSF2008] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. T. Polk. "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile". RFC 5280, 2008.

[D13] Deka, Bhaswati. "Secure Localization Topology and Methodology for a Dedicated Automated Highway System." (2013).

[DGJ2020] D. Derler, K. Gellert, T. Jager, D. Slamanig, C. Striecks. "Bloom Filter Encryption and Applications to Efficient Forward-Secret 0-RTT Key Exchange." In submission, 2020.

[DHT2002] E. M. Dashofy, A. van der Hoek, and R. N. Taylor, "Towards Architecture-based Self-healing Systems," in Proceedings of the First Workshop on Self-healing Systems, ser. WOSS '02. New York, NY, USA: ACM, 2002, pp. 21–26.

[DRS2020] D. Derler, S. Ramacher, D. Slamanig, C. Striecks. "I Want to Forget: Fine-Grained Encryption With Forward Secrecy Meets Decentralization." Cryptology ePrint Archive 2019/912, https://eprint.iacr.org/2019/912. In submission, 2020.

[ERI2016] "Cellular networks for massive IoT: Enabling low power wide area applications," Ericsson, Technical Report, January 2016. https://www.ericsson.com/en/reports-and-papers/white-papers/cellular-networks-for-massive-iot--enabling-low-power-wide-area-applications

[GHJ2017] Guenther, F., Hale, B., Jager, T., Lauer, S.: 0-RTT key exchange with full forward secrecy. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part III. LNCS, vol. 10212, pp. 519–548. Springer, Heidelberg, Germany, Paris, France (Apr 30–May 4, 2017)

[GRKB18] Großwindhager, B., Rath, M., Kulmer, J., Bakr, M. S. A., Boano, C. A., Witrisal, K., & Römer, K. U. (2018). SALMA: UWB-based Single-Anchor Localization System Using Multipath Assistance. in SALMA: UWB-based Single-Anchor Localization System using Multipath Assistance (S. 132-144)

[HJL2017] Hale, B., Jager, T., Lauer, S., Schwenk, J.: Simple security definitions for and constructions of 0-RTT key exchange. In: Gollmann, D., Miyaji, A., Kikuchi, H. (eds.) ACNS 17. LNCS, vol. 10355, pp. 20–38. Springer, Heidelberg, Germany, Kanazawa, Japan (Jul 10–12, 2017)

[HO2013] O. Hoeftberger and R. Obermaisser, "Ontology-based Runtime Reconfiguration of Distributed Embedded Real-Time Systems," in Proc. of the 16th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC 2013), June 2013.

[ESC2016] Elkhodr M, Shahrestani S, Cheung H. Emerging Wireless Technologies in the Internet of Things: A Comparative Study. arXiv preprint, arXiv161100861; 2016.

[EVI2009] Alastair Ruddle, David Ward, Benjamin Weyl, Sabir Idrees, Yves Roudier, Michael Friedewald, Timo Leimbach, Andreas Fuchs, Sigrid Gurgens, Olaf Henniger, et al. Security requirements for automotive on-board networks based on dark-side scenarios. EVITA Deliverable D, 2(3), 2009.

[FGM2015] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," IEEE Communications Surveys Tutorials, vol. 17, no. 4, 2015

[HEA2016] M. Olsson A. Bokesand T. Olovsson H. Broberg et al. M. Islam, C. Sandberg. Deliverable d1.1 needs and requirements.

[HO2013] O. Hoftberger and R. Obermaisser, "Ontology-based Runtime Reconfiguration of Distributed Embedded Real-Time Systems," in Proc. Of the 16th IEEE International Symposium on Object/Component/ServiceOriented Real-Time Distributed Computing (ISORC 2013), June 2013.

[ISA2018] ISA. The 62443 series of standards: Industrial automation and control systems security. (1–4), 2018.

[KKA2017] Adi Karahasanovic, Pierre Kleberger, and Magnus Almgren. Adapting threat modelling methods for the automotive industry. In Proceedings of the 15th ESCAR Conference, pages 1–10, 2017.

[MLM2006] C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, R. Metz, and B. A. Hamilton, "Reference Model for Service Oriented Architecture 1.0," available at https://www.oasis-open.org/standards\#soa-rmv1.0, OASIS standard.

[MLY2005] Suvda Myagmar, Adam J Lee, and William Yurcik. Threat modelling as a basis for security requirements. In Symposium on requirements engineering for information security (SREIS), volume 2005, pages 1–8. Citeseer, 2005.

[MSCH2016] Zhendong Ma and Christoph Schmittner. Threat modelling for automotive security analysis. Advanced Science and Technology Letters, 139:333–339, 2016.

[OMG2012] Object Management Group, "Service Oriented Architecture Modelling Language," available at http://www.omg.org/spec/SoaML/, OMG Specifiation SoaML.

[OMT2008] P. Oreizy, N. Medvidovic, and R. N. Taylor, "Runtime Software Adaptation: Framework, Approaches, and Styles," in Companion of the 30th International Conference on Software Engineering, ser. ICSE Companion '08. New York, NY, USA: ACM, 2008, pp. 899–910.

[RES2018] E. Rescorla. "The Transport Layer Security (TLS) Protocol Version 1.3." RFC 8446, 2018.

[ROS2009] Matthew Rosenquist. Prioritizing information security risks with threat agent risk assessment. Intel Corporation White Paper, 2009.

[SC2016] Sanchez-Iborra R, Cano M-D. State of the art in LP-WAN solutions for industrial IoT services. Sensors. 2016;16(5):708

[SCH2005] D. C. Schmidt, A. Corsaro, and H. van't Hag, "Addressing the Challenges of Tactical Information Management in Net-Centric Systems with DDS," CrossTalk spec. issue on Distributed Software Development, pp.24–29, May 2005.

[STHS2019] Christoph Schmittner, Peter Tummeltshammer, David Hofbauer, Abdelkader Magdy Shaaban, Michael Meidlinger, Markus Tauber, Arndt Bonitz, Reinhard Hametner, and Manuela Brandstetter. Threat modelling in the railway domain. In International Conference on Reliability, Safety, and Security of Railway Systems, pages 261–271. Springer, 2019.

[SSG2019] Abdelkader Magdy Shaaban, Christoph Schmittner, Thomas Gruber, A Baith Mohamed, Gerald Quirchmayr, and Erich Schikuta. Ontology-based model for automotive security verification and validation. In Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services, pages 73–82, 2019.

[SKS2018] Abdelkader Magdy Shaaban, Erwin Kristen, and Christoph Schmittner. Application of IEC62443 for IoT components. In International Conference on Computer Safety, Reliability, and Security, pages 214–223. Springer, 2018.

[SWG2014] D. Schmidt, J. White, and C. Gill, "Elastic Infrastructure to Support Computing Clouds for Large-Scale Cyber-Physical Systems," in Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC), 2014 IEEE 17th International Symposium on, June 2014, pp.56–63

[SW09] Stefan Saroiu and Alec Wolman. 2009. Enabling new mobile applications with location proofs. In Proceedings of the 10th workshop on Mobile Computing Systems and Applications (HotMobile ,09). ACM, New York, NY, USA, Article 3, 6 pages. DOI=http://dx.doi.org/10.1145/1514411.1514414

[TCB15] Manoop Talasila, Reza Curtmola, Cristian Borcea, Collaborative Bluetooth-based location authentication on smart phones, Pervasive and Mobile Computing, Volume 17, Part A, 2015, pages 43-62, ISSN 1574-1192.

[VMAJ2019] Omar Veledar, Georg Macher, Eric Armengaud, Stefan Jaksic, Christoph Schmittner, Violeta Damjanovic-Behrendt, Christos Thomos, Kay Uwe Romer, Konrad Diwold, Leo Happ Botler, Mario Drobics, and Eva Maria Holzer. Safety and security of iot-based solutions for autonomous driving: Architectural perspective. 2019. 6th International Symposium on Model-Based Safety and Assessment, IMBSA 2019; Conference date: 16-10-2019 Through 18-10-2019.

[VVC2018] „Choosing IoT-connectivity? A guiding methodology based on functional characteristics and economic considerations.", Vannieuwenborg, Frederic, Sofie Verbrugge, and Didier Colle. Transactions on Emerging Telecommunications Technologies 29, no. 5 (2018): e3308.

[ZG2013] Kai Zhao and Lina Ge. A survey on the internet of things security. 2013 9th international conference on computational intelligence and security, pages 663–667. IEEE, 2013.

For more information on this document or the IoT4CPS project, please contact:
Mario Drobics, AIT Austrian Institute of Technology, mario.drobics@ait.ac.at

Layout & Grafik

Nora Novak, goldmaedchen Grafikdesign

The IoT4CPS project is partially funded by the „ICT of the Future“ Program of the FFG and the BMVIT.