Resilience of Virtualized Embedded IoT Networks

Doğanalp Ergenç, Mathias Fischer
University of Hamburg
{ergenc, mfischer}@informatik.uni-hamburg.de

***Abstract*—Embedded IoT networks are the backbone of safety-critical systems like smart factories, autonomous vehicles, and airplanes. Therefore, their resilience against failures and attacks should be a prior concern. The design of more capable IoT devices enables the flexible deployment of network services by virtualization but it also increases the complexity of the systems and makes them more error-prone. In this paper, we discuss the issues and challenges to ensure resilience in virtualized embedded IoT networks by presenting proactive and reactive measures.**

***Index Terms*—embedded systems, IoT, resilience, virtualized**

I. Introduction

Embedded systems as used in autonomous vehicles, airplanes, and industrial networks have become complex ecosystems. For instance, the latest Tesla autopilot is supported by eight cameras and twelve ultrasonic sensors. Similarly, with Industry 4.0 intelligent cyber-physical systems emerge that are composed of a multitude of collaborating embedded devices hosting mission-critical services. We currently observe that trends from conventional computer networks, like more powerful devices and virtualization, are widely adopted in the (embedded) IoT domain [1]. As a result, these systems can take over more complex tasks and can operate multiple virtualized services on top of a physical node. It provides significant flexibility to deploy and migrate IoT services over the network. For example, an automatic braking system in an autonomous car can be designed as a chain of services as shown in Fig. 1. First, a group of sensor-connected image processing nodes detects objects on the road (as a service). Second, another group of nodes can initiate an automatic emergency braking based on the information received from the first group. The same functionality, e.g., image processing, can be migrated to other candidate nodes when using virtualization. However, it also increases the complexity of the systems, making them more error-prone and vulnerable. Especially safety-critical embedded systems should be resilient as much as possible against network failures and targeted attacks. Resilience can bridge the safety and security domains to maintain a system's correct services in case of failures and attacks, to provide graceful degradation in worst-case, and to recover to a normal system state [2]. The contribution of this paper is the discussion of issues and challenges for (i) the design of such a heterogeneous embedded IoT network without compromising neither service quality nor resilience and (ii) the active protection of that via required mechanisms against failures and attacks. In the rest of the paper, we discuss the issues of the proactive and reactive measures to design and maintain a resilient embedded IoT network in Sections II and III, respectively.

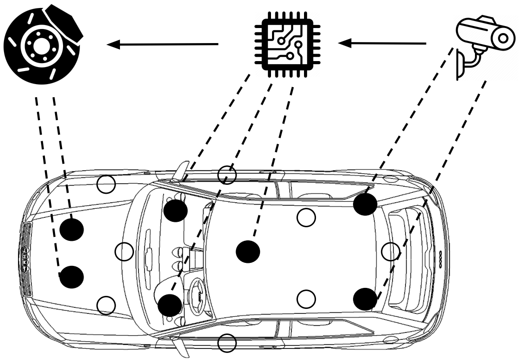

Fig. 1. Different services, e.g., cameras, image processing, and automatic breaking, can be activated on some embedded units (black nodes) and migrated to the others (white nodes) in case of failures.

II. Issues of Proactive Measures

Proactive measures are considered through the design stage to increase (or guarantee) service availability. Adding resilience by redundancy e.g., hot/cold backups and design diversity, is usually considered in safety-critical systems so far. However, their implementation, management, and synchronization induce extra burden. Moreover, they can maintain the availability of a system only in the presence of specific threats like Byzantine failures. Therefore, further proactive measures should be considered as discussed in the following.

Time-sensitive nature of services. Missing a deadline in a time-sensitive service is considered as a failure with potentially devastating impact. The deployment and provision of services and redundant resources should be accordingly designed to fulfill the deterministic communication requirements. Due to such requirements, it is not sufficient to guarantee only the availability of inter-service communication with traditional resilience measures, but strict determinism should also be maintained. Thus, the time-sensitive nature of the embedded IoT services additionally complicates the resilient design.

Spare resources and dimensioning. Various proactive measures require extensive resource planning since a network and its nodes should guarantee sufficient resources to be able to maintain the service availability in case of failures. For instance, to enable service and traffic migrations [3], or add redundant resource e.g., hot-backups, such planning is vital. Further strategic choices, e.g., spatial placement of redundant instances to encourage local repairs, also need a careful organization. The complexity of planning spare resources and network dimensioning significantly increases even for small-sized networks under simple resilience constraints like fault-tolerance against only single node failures. However, optimum resource planning considering further failure and attack scenarios is a more challenging problem.

Heterogeneity of service characteristics. The heterogeneity in service characteristics straitens design choices. While some services need to be distributed over the network via multiple instances, others have a few instances if not only one. If resources are spare, e.g., as a result of an attack or fault, mission-critical services should have a higher priority in occupying network resources, while others should be served on a best-effort basis only. Even the subsystems of highly mission-critical systems such e.g., parts of an aircraft, have separate and well-defined resilience requirements. As a result, an increasing heterogeneity in service characteristics adds extra design parameters and complicates the resilient system design.

Shaping a fault-hypothesis. Mission-critical networks should have the utmost resilience against any sort of threat to avoid catastrophic consequences. There may be mass failures due to software crashes, power cuts, or accidents. Cyber-attacks can be targetted to specific hardware or software components, and turn to be epidemics affecting a multitude of network elements. Fault-hypothesis is the assumption concerning the types, risks, and number of faults that a system is expected to withstand. In the design phase, the fault-hypothesis should be well-defined and the tradeoff between QoS-optimality, cost, e.g., for redundancy, design, and implementation time, and resilience should be considered. Moreover, the hypothesis should realize the difference between service resilience, e.g., availability of services, and network resilience i.e., availability of network elements and nodes, and guarantee the maintenance of the former.

III. Issues of Reactive Measures

Proactive measures enable a networked system to react when an error occurs due to a failure or attack. We also need reactive measures to detect and mitigate failures, and monitor the status of the services via resilience mechanisms such as Monitoring and Control Systems (MaCSs). The required characteristics of MaCS to satisfy requirements of embedded IoT networks are discussed in this section.

Time-sensitive detection. Apart from missing an error or attack, even a slightly delayed detection may be disruptive for time-sensitive services. Any disrupting attack or failure should be detected and the required countermeasure should be applied as early as possible right after the detection. Therefore, we need more intelligent e.g., high-precision and predictive, detection and mitigation algorithms, and proper MaCS architecture, e.g., encouraging quick and local mitigation/failover to maintain inter-service deterministic communication.

Fault forecasting. Fault forecasting helps to predict a possible failure in advance, especially for the systems whose general behavior is well-defined. Mission-critical systems have been usually isolated via strict access control schemes and thus anomalies have been observable with strong indicators. However, modern embedded IoT systems are highly heterogeneous and connected ecosystems with mixed safety-critical and non-safety-critical services. Therefore, a more tedious and holistic behavioral analysis is required on service behavior and inter-service relationships for accurate forecasting.

Noncentralized architecture. A centralized MaCS may provide a network-wide view and higher precision in detecting failures. However, it has scalability issues and may have a single point of failure depending on the failure scenarios. Moreover, a centralized MaCS suffers from additional communication delays, while a local and decentralized MaCS could detect and react upon failures and attacks close to their origin and promote quick failovers. When we consider the failure tolerance of safety-critical systems from the order of milliseconds to nanoseconds, a proper architecture that minimizes the detection and mitigation delay is required.

Consensus and synchronization. Even if noncentralized architectures are promising to address the time-sensitivity, it brings some complexities such as consensus and synchronization between MaCS instances. A distributed MaCS promotes local provisioning and repairs, but still a consensus is required to ensure the overall operating state of the network after any change e.g., service migration. Moreover, the cooperation between MaCS may be required to make globally optimal decisions where a local decision may not be sufficient or feasible. For instance, they should ensure the end-to-end traffic is established within a bounded delay even if they are not directly responsible for route finding. Most importantly, the definition and prioritization of the cases that require consensus in real-time or long-term are crucial to implement a consensus and synchronization scheme.

IV. Conclusion

Embedded IoT networks constitute various modern safety-critical systems. Their resilience against attacks and failures should be prior design concerns to avoid the loss of life and property. In this paper, we discuss the main issues of a resilient embedded IoT network design under two categories, the issues for proactive and reactive resilience measures to envision further research questions. In those categories, we summarize the important points to increase the robustness of the system by design without sacrificing QoS and resilience via the design of monitoring and controlling instruments, respectively.

References

[1] C.-S. Shih, J.-J. Chou, and K.-J. Lin, “WuKong: Secure Run-Time environment and data-driven IoT applications for Smart Cities and Smart Buildings,” tech. rep.

[2] J. Rak, *Resilient Routing in Communication Networks (Springer)*. 11 2015.

[3] K. Ogawa, H. Sekine, K. Kanai, K. Nakamura, H. Kanemitsu, J. Katto, and H. Nakazato, “Performance evaluations of iot device virtualization for efficient resource utilization,” in *2019 Global IoT Summit (GIoTS)*, pp. 1–6, June 2019.