Toward a Security Architecture for Multi-Embedded-Agent Systems

1st Arthur Baudet
LCIS
Univ. Grenoble Alpes, Grenoble INP
Valence, France
arthur.baudet@lcis.grenoble-inp.fr

2nd Oum-El-Kheir Aktouf
LCIS
Univ. Grenoble Alpes, Grenoble INP
Valence, France
oum-el-kheir.aktouf@lcis.grenoble-inp.fr

3rd Annabelle Mercier
LCIS
Univ. Grenoble Alpes, Grenoble INP
Valence, France
annabelle.mercier@lcis.grenoble-inp.fr

4th Philippe Elbaz-Vincent
Institut Fourier
Univ. Grenoble Alpes, CNRS
F-38000 Grenoble, France
philippe.elbaz-vincent@univ-grenoble-alpes.fr

***Abstract*—In this paper we establish the context of multi-embedded-agent systems, a kind of decentralized systems with autonomous embedded devices communicating and cooperating to achieve a common goal, and wide spectrum of threats that are posed to them as such systems can be vulnerable to attacks coming from the inside the system as of the outside and through a wide range of vectors from hardware to software weaknesses. We then try to establish a base of what would be a global security architecture for multi-embedded-agent systems to finally focus on problems rising from the use of cryptography (especially public key cryptography) in a decentralized environment.**

***Index Terms*—Multi-agent systems, embedded agent, security architecture, decentralized security**

I. Introduction

Multi-embedded-agent system (MEAS) design is a paradigm where multiple embedded devices (called embedded agents) act in a decentralized manner, collaborating to achieve their goals and leading to the achievement of the system main goal. Moreover, the agents are autonomous, they can react to their environment or learn from their previous experiences [1]. MEAS allow the control and coordination of large distributed systems such as mobile area networks, wireless sensor networks or drone fleets with any large number of devices since the computation and data are distributed throughout the system, with no single point of failure or bottleneck.

However, the absence of a decentralized authority and the need for cooperation also widen the attack surface of those systems as they are not only vulnerable to hardware, network and software attacks but also specific attacks carried out by malicious, intruding agents. To avoid such situations, these MEAS need to be secured by a global security solution, supported by a global security architecture. The objective of this work is to find out the most relevant and important components of such a global security architecture and to show how they articulate with the secured MEAS.

In the following, we will first describe the main attacks that can be used against MEAS and the common countermeasures for those attacks. Then, we will focus on the challenges raised by the use of cryptography in a completely decentralized environment. Finally, most likely solutions we plan to study in our future work in order to meet those challenges are introduced.

II. Threats in multi-embedded-agent systems

In this section, the main threats against MEAS are described according the the architectural level or function of MEAS: hardware, communication, and agent interaction. Starting at the hardware level, as embedded agents are concerned more than software ones, they should be protected against any physical intrusion into the hardware, mainly any side-channel attack or fault injection attack. MEAS should also be able to detect jamming attacks or denial of service attacks on their communications that are, more often than not, wireless. Also because the communications are wireless and thus easily accessible, they should be protected against man-in-the-middle and man-on-the-side attacks. Still about the communication but this time from a routing point of view, as most MEAS rely on ad hoc networks (*e.g.*, wireless sensor networks or mobile area networks), routing is done dynamically by specific algorithms. Consequently, MEAS may experience weaknesses against routing attacks such as sinkhole attacks, wormhole attacks, black hole attacks or gray hole attacks. Moving to more specific attacks, enabled by the communication between agents, the cooperation between agents can be abused by corrupted agents to disrupt the system. Moreover, MEAS may be designed to integrate new agents at run-time, for example a new device in a network of embedded devices in a smart home. Thus, an attacker can try to add a malicious device behaving as an agent but with malicious intent. Even if the MEAS wasn't designed to accept new agents, it may not be possible to distinguish between new or old agents leading to a Byzantine attack model where one agent cannot assess the

This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02)

states and intentions of the other agents before interacting with them. Depending on the actual chosen system, all attacks will not be possible and more specific attacks may be added, but, in order to provide a general security framework, we define the following threat model: (i) agents are physically accessible, (ii) communications are wireless and ad hoc with no prerequisite on the wireless medium characteristics, and (iii) from an agent point of view, any other agents are modeled as Byzantine nodes. Furthermore, (iv) as agents are embedded devices, they are limited in resources but not to the point where any security-related computation are impossible since this would prevent any attempt on securing them.

III. First steps toward a security architecture

Following the saying “a chain is only as strong as its weakest link” (Thomas Reid), a MEAS can only be said to be secured if every part of the system is protected.

And, as all computation are done on hardware, hardware should be the first part to be secured. A trust anchor [2] should be established so every security can be derived from this anchor. Protecting the most critical computation requires specific hardware such as secure elements or trust platform modules. Depending on the quality of the chip, the trust anchor should prevent most of the side channel and fault injection attacks from revealing key data of the equipped agent. Jamming attacks cannot be fully countered but can at least be detected by several ways of listening to the communication medium.

Communication could be protected using cryptography but, as we will discuss later, its use is limited because of the fully decentralized context of MEAS.

Most of application level security threats, such as attacks on routing or cooperation, can be countered by a trust scheme or an intrusion detection system (even though intrusion detection systems tend to be centralized and require execution traces, some are adapted to run-time detection in MEAS). There are extensive works on trust solutions [3] and we cannot recommend one specific implementation for all systems but rather to choose the most adequate to each specific use case.

However, all the solutions given above come at a cost. Either a monetary cost to add a specific security chip or resource to encrypt and decrypt data, compute trust values, continually listen to the communication medium, store trust information, *etc*. But the trade-off between high security and lower cost exists in any information system and ultimately, whatever the recommendation we make, budgetary and sometimes implementation constraints may hinder the application of a complete security solution.

IV. The challenge of cryptography in a decentralized context

Several authors consider that cryptography-based solutions cannot be used in decentralized systems as they lack a central authority to manage the cryptographic keys of the agents (distribution, sharing or revocation) [4]. Indeed, the most common infrastructure, the Public Key Infrastructure, requires a hierarchy or third parties to certify the authenticity of a source.

Blockchain Technologies have been studied to provide solutions to this issue as they are a way to solve the consensus problem in decentralized systems [5]. However, their most studied implementation, the proof of work, is very inadequate for use with embedded agents as it will deplete all their energy in a very short time. A second limitation is that a blockchain can only grow and its memory overhead can also be prohibiting. Still, work is done to lighten blockchain technologies with the aim to use them with embedded devices [6].

Even if relevant and interesting blockchain technologies could be found, they still require the use of pre-established pairs of public and private keys and we would think that it is desirable to allow agents to generate keys at run-time if, when entering the system, none were provided before. To this end, we are currently studying the use of identity- and attribute-based encryption. Both cryptographic schemes require a central authority to provide a master key but current research is done to find decentralized schemes [7].

And so, we expect that contributing on expanding the use of attribute-based encryption to MEAS could be beneficial for both research on securing MEAS and attribute-based encryption.

V. Conclusion and future work

In this paper, we first gave a definition of MEAS, presented common threats to their security, and relevant countermeasures to those threats. We then focused on using cryptography in this context as its use is paramount to provide confidentiality, integrity and authentication. We explained why existing cryptography schemes are not relevant in decentralized systems. Finally, to solve this issue, we presented two main solutions: blockchain technologies and identity- and attribute-based encryption, both having to be adapted before being deployed on MEAS.

Our ongoing work is studying the use of attribute-based encryption in MEAS to provide run-time, decentralized key management capabilities to the agents and pave the way to a global security architecture for MEAS.

References

[1] M. Wooldridge and N. R. Jennings, “Intelligent agents: theory and practice,” *The Knowledge Engineering Review*, 1995.

[2] F. Brasser, B. El Mahjoub, A.-R. Sadeghi, C. Wachsmann, and P. Koeberl, “Tytan: Tiny trust anchor for tiny devices,” in *Proceedings of the 52nd Annual Design Automation Conference*, 2015.

[3] J.-H. Cho, K. Chan, and S. Adali, “A survey on trust modeling,” *ACM Comput. Surv.*, 2015.

[4] R. H. Jhaveri and N. M. Patel, “Attack-pattern discovery based enhanced trust model for secure routing in mobile ad-hoc networks,” *International Journal of Communication Systems*, 2017.

[5] D. Calvaresi, A. Dubovitskaya, J. P. Calbimonte, K. Taveter, and M. Schumacher, “Multi-agent systems and blockchain: Results from a systematic literature review,” in *Advances in Practical Applications of Agents, Multi-Agent Systems, and Complexity: The PAAMS Collection*, 2018.

[6] S. Falcone, J. Zhang, A. Cameron, and A. Abdel-Rahman, “Blockchain design for an embedded system,” *Ledger*, 2019.

[7] T. OKAMOTO and K. TAKASHIMA, “Decentralized attribute-based encryption and signatures,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 2020.