# Remote Attestation of Heterogeneous Cyber-Physical Systems: The Automotive Use Case (Extended Abstract)

Karim El Defrawy
HRL Laboratories
kmedefrawy@hrl.com

Gavin Holland
HRL Laboratories
gdholland@hrl.com

Gene Tsudik
UC Irvine
gts@ics.uci.edu

***Abstract*—Cyber-Physical Systems (CPS) are increasingly permeating our daily lives, particularly in the automotive domain since a modern vehicle can be regarded as one complex CPS. Given their increasing importance, CPS (and automotive systems as a representative case) are becoming attractive targets for attacks. Several techniques with varying assumptions and limitations have been proposed to detect and/or mitigate such attacks. A common theme has been the need for Remote Attestation (RA), a security service that allows a trusted party (verifier) to check the internal state of a remote untrusted and possibly compromised system (prover).**

**This talk provides a first stab at extending contemporary RA techniques to settings with heterogeneous CPS. This is in contrast to settings with standalone or single devices which have been the focus of existing research. We propose to efficiently and securely combine the attestation of multiple devices thus providing a (natural) security service for larger and more complex CPS. We focus on the automotive domain and investigate how to realize attestation of multi-device CPS. We conclude with a discussion of future research directions and open problems surrounding RA in the automotive and general CPS domains.**

## I. Introduction

According to the U.S. Networking and Information Technology Research and Development (NITRD) Program Cyber-Physical Systems (CPS) can be defined as[1]: *"Cyber Physical Systems (CPS) are smart networked systems with embedded sensors, processors and actuators that are designed to sense and interact with the physical world (including the human users), and support real-time, guaranteed performance in safety-critical applications..."*.

We argue that (most) CPS operate in restricted environments and with a potentially large (but enumerable) set of inputs and outputs, and a relatively small state space, and maybe even simple control functions and logic, compared to general purpose computing systems. Such a restricted state space enables attestation of CPS components to ensure that no component is behaving maliciously. Another reasonable restriction, especially in the automotive use case, is that most worrisome attacks are conducted remotely; physical attacks on CPS do not scale. If physical access is allowed, there's little one can do without special tamper resistant hardware. Our focus is on remote attacks.

[1]http://www.nitrd.gov/nitrdgroups/images/6/6a/Cyber_Physical_Systems_%28CPS%29_Vision_Statement.pdf

*Remote Attestation (RA)* has emerged as a promising security service to fortify embedded systems against remote attacks and to allow one to establish a static or dynamic root of trust [1]. We argue that RA can also play a pivotal role in securing CPS against malware and remote attacks. In a typical RA setting a (trusted) verified will challenge an (untrusted) prover, and require the prover to provide unforgaeble evidence that it is in a legitimate state, otherwise if this is not the case, the prover will detect any cheating attempts. If the state covers the program and data memory, the verifier is ensured that no malware is resident on the embedded device.

*This talk represents a first stab at extending existing RA techniques for (single) embedded devices to the larger and more complex CPS. We focus on the automotive use case as a concrete CPS setting.*

## II. Outline of Proposed Talk

This talk has three main parts: (1) overview of (recent) research on remote attestation, (2) attestation of heterogeneous CPS (with a focus on automotive systems), and (3) future challenges and research problems.

1) **Overview of Research in Remote Attestation**
   The first part overviews existing research (including our own) in remote attestation and covers:
   a) *Formal Models and Definitions of Remote Attestation:* There is little work formalizing various notions of remote attestation; recently, [2] provided a security framework to design software attestation schemes, and [3] provided a game-based definition that bears some resemblance to a Message Authentication Code (MAC)-Forge game. These models and definitions mainly focus on standalone verifiers. We believe that it is uncertain whether they are suitable for more complex CPS.
   b) *Software-Only Remote Attestation:* In general, software-only attestation proposals (e.g., [4]) rely on strong assumptions on adversarial capabilities and only work if the verifier communicates directly to the prover, with no intermediate hops. Most software-only schemes compute a checksum over the memory of the device using a specially-crafted function that includes side-effects in its computation, such that any emulation of this function incurs a sufficient delay that enables detecting any cheating attempts. Some

assumptions that form the basis for these solutions have been challenged in [5] and several attacks on software-only schemes have been demonstrated in [6]. Moreover, [7] showed that time-based attestation schemes may be vulnerable to Time Of Check, Time Of Use (TOCTOU) attack.

The above limitations render software-only attestation approaches inadequate for attestation performed over a network and thus of limited applicability to the automotive setting and other CPS. An interesting question is whether such approaches can be utilized by CPS components to attest their low-end counterparts within the same CPS.

c) *Hardware-Based Remote Attestation:* There are some commercial and standardized techniques for attestation using secure hardware, e.g., ARM TrustZone [8] and Trusted Platform Module (TPM) based [9] high-end microprocessors [10]. Other research proposals include SPM [11] which provides hardware-based mechanism for process isolation. SPM relies on a special vault module bootstrapped from a static root of trust. [12] proposes a hardware architecture that provides a dynamic root of trust and application isolation and authentication. In [13], a *Logic for Secure Systems (LS2)* that rely on a TPM is proposed. LS2 is used to specify a remote attestation protocol standardized by TCG but without providing an explicit definition of remote attestation. We revisit *LS2* in the attestation of CPS and future research portions of this talk.

d) *Hybrid Remote Attestation:* [1], [3], [14] focused on the design space between the two extremes of software-only and hardware-based RA. [1] develops a new primitive (called SMART) based on hardware-software co-design for securely establishing a dynamic root of trust in a remote embedded device. SMART focuses on low-end micro-controller units (MCU) that lack specialized memory management or protection features, it requires minimal changes to existing MCUs (while providing concrete security guarantees) and assumes few restrictions on adversarial capabilities. [3] and [14] define a new security notion for remote attestation protocols and specify necessary and sufficient properties needed for a device to support secure remote attestation. We review both [1] and [3], [14] in detail and examine their applicability to more generic CPS.

2) **Attestation of Heterogeneous CPS:** In this portion of the talk we consider extending current definitions and models of remote attestation to heterogeneous CPS. We examine if the order of attesting different components affects security guarantees of the overall (combined) process. We then follow with a discussion of an automotive system as a concrete example of a complex CPS to be attested.

a) *Models, Definitions and Interfaces:* We examine whether current remote attestation definitions such as [2] and [3] can be extended to allow composability. To that end, we explore whether certain security modeling frameworks used in the secure computation literature (e.g., the Universal Composability (UC) Framework [15] and variants thereof) may be of relevance in the setting of heterogeneous CPS. We also discuss the role that a logical framework such a LS2 [13] can play in reasoning and proving security properties (of designed protocols, techniques and primitives) in heterogeneous CPS.

b) *Order and Time:* The order of attesting different components in a CPS is important to guarantee security; one can imagine a malware that moves around components to avoid detection if such components are attested at different times. A simple solution would be to attest all components at the same time. However, this may not be possible in most CPS because of the natural need to minimize the overall time devoted to attestation (since performing attestation takes the entire CPS away from its central tasks). It is unclear whether such a short window would be enough to attest components with large memory (e.g., $100s$ of KB). For example, SMART[3] requires computing a hash-based Message Authentication Code (HMAC) of the memory space which took $287ms$ to execute on a low-end MCU at $8MHz$. Scheduling components to attest while guaranteeing minimal interruption to a CPS is an interesting open problem.

c) *Automotive Use Case:* Modern automotive systems are complex CPS with over 70 Electronic Control Units (ECU) and several wireless interfaces all open to attacks and malware. In this part of the talk we consider how to build an In-vehicle Root of Trust (IRT) that establishes and manages keys and performs remote attestation of un-trusted vehicle modules. We argue that ECU in automotive systems can be classified into three categories, (1) high-end, (2) medium-end, and (3) low-end. *High-end modules* may contain some hardware support for RA (e.g., ARM TrustZone [8]). A vehicle may contain one or two such modules which can serve as the IRT. *Medium-end modules* may be commercial micro-controllers such as AVR [16] and MSP [17], which can be modified to implement SMART as demonstrated in [1]. Alternatively such modules may be based on work in the E-safety Vehicle Intrusion Tolerant Applications (EVITA) project [18] which introduced Hardware Security Module (HSM) to ECU to implement cryptographic primitives, such as key generation and management. *Low-end modules* are the simplest and cheapest available controllers and may not have any hardware support for attestation. They also have limited memory and speed, e.g., pressure and tire sensors. The IRT or a medium-end module may attest their memory (assuming it has access to it, alternatively software-based attestation may be added to their firmware). Given these three categories of components, the IRT can attest all modules in the vehicle at startup time (or boot-time to borrow language from the computing realm) thus establishing a static root-of trust. This process may take a relatively long time (e.g., several seconds). On the other hand, since a vehicle takes several seconds to warm up, this delay might be acceptable for current and future consumers. To establish a dynamic-root of trust, the IRT has to randomly attest components while its running, without interfering with safety or correct operation. The challenge is how to devise a strategy to allow the IRT and medium-end components to attest other medium and/or low-end components. A more interesting question is

what should be the response of the IRT once a module fails attestation. These issues and others are discussed in this talk.

3) **Future Challenges and Research Directions:**
Recent remote attestation surveys such as [19] briefly mention attestation of heterogeneous embedded devices. However, they mainly focus on low-end embedded devices. [19] mentions three important challenges in designing remote attestation techniques: (1) the minimal set of hardware and software features for a low-end embedded device to attain provably-secure remote attestation, (2) mechanisms and features needed to protect the remote attestation process from unauthorized invocation, and (3) how to design efficient remote attestation for a multitude of potentially heterogeneous embedded devices, e.g., within a vehicle, a household, or an aircraft.
The last portion of this talk looks into challenges in extending remote attestation to heterogeneous CPS with a mix of high-end and low-end capabilities. It will also examine the missing features in existing models and definitions. We utilize an automotive system as a case study and overview how one may use an extended version of the *LS2* logic to reason about attestation in such a system. The talk ends with laundry-list of concrete research challenges and problems to solve to realize remote attestation of heterogeneous CPS. The main goal of this part of the talk is to solicit feedback from ESCAR participants on the proposed research directions.

*In summary, this talk extends the discussion started in [19] by considering more generic CPS that may have more processing and memory than low-end devices which were considered before. We will also consider potential formal models and frameworks to reason about security of RA in CPS, with a special focus on composability, we will then finally conclude with a brief discussion of potential automated verification and synthesis of RA protocols and primitives.*

## REFERENCES


[1] K. Eldefrawy, A. Francillon, D. Perito, and G. Tsudik, "SMART: Secure and Minimal Architecture for (Establishing a Dynamic) Root of Trust," in *NDSS 2012, 19th Annual Network and Distributed System Security Symposium, February 5-8, San Diego, USA*, San Diego, UNITED STATES, 02 2012. [Online]. Available: http://www.eurecom.fr/publication/3536

[2] F. Armknecht, A.-R. Sadeghi, S. Schulz, and C. Wachsmann, "A security framework for the analysis and design of software attestation," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, ser. CCS '13. New York, NY, USA: ACM, 2013, pp. 1–12. [Online]. Available: http://doi.acm.org/10.1145/2508859.2516650

[3] A. Francillon, Q. Nguyen, K. B. Rasmussen, and G. Tsudik, "Systematic treatment of remote attestation," Cryptology ePrint Archive, Report 2012/713, 2012, http://eprint.iacr.org/.

[4] A. Seshadri, M. Luk, E. Shi, A. Perrig, L. van Doorn, and P. Khosla, "Pioneer: Verifying code integrity and enforcing untampered code execution on legacy systems," *SIGOPS Oper. Syst. Rev.*, vol. 39, no. 5, pp. 1–16, Oct. 2005. [Online]. Available: http://doi.acm.org/10.1145/1095809.1095812

[5] U. Shankar, M. Chew, and J. D. Tygar, "Side effects are not sufficient to authenticate software," in *Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13*, ser. SSYM'04. Berkeley, CA, USA: USENIX Association, 2004, pp. 7–7. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251375.1251382

[6] C. Castelluccia, A. Francillon, D. Perito, and C. Soriente, "On the difficulty of software-based attestation of embedded devices," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 400–409. [Online]. Available: http://doi.acm.org/10.1145/1653662.1653711

[7] X. Kovah, C. Kallenberg, C. Weathers, A. Herzog, M. Albin, and J. Butterworth, "New results for timing-based attestation," in *Security and Privacy (SP), 2012 IEEE Symposium on*, May 2012, pp. 239–253.

[8] "Arm trustzone specifications," 2015, http://www.arm.com/products/processors/technologies/trustzone/index.php.

[9] "Trusted computing group (tcg) trusted platform module (tpm) main specification," 2015, http://www.trustedcomputinggroup.org/resources/tpm_main_specification.

[10] "Intel trusted platform module (tpm module-axxtpme3) hardware users guide," 2015, http://download.intel.com/support/motherboards/server/sb/g21682003_tpm_hwug.pdf.

[11] R. Strackx, F. Piessens, and B. Preneel, "Efficient Isolation of Trusted Subsystems in Embedded Systems," in *SecureComm*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, S. Jajodia, J. Zhou, S. Jajodia, and J. Zhou, Eds., vol. 50. Berlin, Heidelberg: Springer, 2010, pp. 344–361. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-16161-2_20

[12] J. Noorman, P. Agten, W. Daniels, R. Strackx, A. Van Herrewege, C. Huygens, B. Preneel, I. Verbauwhede, and F. Piessens, "Sancus: Low-cost trustworthy extensible networked devices with a zero-software trusted computing base," in *Proceedings of the 22Nd USENIX Conference on Security*, ser. SEC'13. Berkeley, CA, USA: USENIX Association, 2013, pp. 479–494. [Online]. Available: http://dl.acm.org/citation.cfm?id=2534766.2534808

[13] A. Datta, J. Franklin, D. Garg, and D. Kaynar, "A logic of secure systems and its application to trusted computing," in *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, ser. SP '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 221–236. [Online]. Available: http://dx.doi.org/10.1109/SP.2009.16

[14] A. Francillon, Q. Nguyen, K. B. Rasmussen, and G. Tsudik, "A minimalist approach to remote attestation," in *Proceedings of the Conference on Design, Automation & Test in Europe*, ser. DATE '14. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2014, pp. 244:1–244:6. [Online]. Available: http://dl.acm.org/citation.cfm?id=2616606.2616905

[15] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," in *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, ser. FOCS '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 136–. [Online]. Available: http://dl.acm.org/citation.cfm?id=874063.875553

[16] "Atmel avr 8-bit and 32-bit microcontrollers," 2015, http://www.atmel.com/products/microcontrollers/avr/.

[17] "Overview for low-power microcontrollers," 2015, http://www.ti.com/lsds/ti/microcontrollers_16-bit_32-bit/msp/overview.page.

[18] "E-safety vehicle intrusion protected applications (evita)," 2015, http://www.evita-project.org/publications.html.

[19] G. Tsudik, "Challenges in remote attestation of low-end embedded devices," in *Proceedings of the 4th International Workshop on Trustworthy Embedded Devices*, ser. TrustED '14. New York, NY, USA: ACM, 2014, pp. 1–1. [Online]. Available: http://doi.acm.org/10.1145/2666141.2668383