Poster: LIghtweight Swarm Attestation: a Tale of Two LISA-s

Xavier Carpent[1], Karim ElDefrawy[*2], Norrathep Rattanavipanon[1], and Gene Tsudik[1]

[1]Computer Science Department, University of California, Irvine
[2]Computer Science Lab, SRI International
xcarpent@uci.edu, karim@csl.sri.com, {nrattana, gene.tsudik}@uci.edu

***Abstract*—The advent of IoT brought about the paradigm of many interconnected devices, thus triggering the need for efficient collective attestation of a (possibly mobile) swarm of provers. Though recent work has yielded some initial concepts for swarm attestation, several key issues remain unaddressed, and practical realizations have not been explored. This paper aims at advancing swarm attestation by bringing it closer to reality. Towards that goal, the paper makes two contributions: (1) defines a new metric, called QoSA: Quality of Swarm Attestation, that captures the type of information offered by a swarm attestation technique, and (2) constructs two practical attestation protocols with different QoSA features and communication and computation complexities. Security of proposed protocols is analyzed and their performance is assessed based on experiments with prototype implementations.**

I. Introduction

In the last decade, Remote Attestation (RA) emerged as a distinct security service with the main goal of detecting malware presence on embedded systems and IoT devices. Essentially, RA is the process whereby a trusted entity (verifier or $\mathcal{V}$rf) securely verifies current internal state of an untrusted and possibly compromised remote device (prover or $\mathcal{P}rv$). RA can help establish a static or dynamic root of trust in the prover. It can also be used as a building block for constructing more specialized security services, such as software updates as well as secure deletion and device resetting.

Various RA techniques with different assumptions, security features and complexities, have been proposed for the single-prover scenario. Nonetheless, new issues emerge when there is a need to attest a potentially large group or swarm of devices. First, it is inefficient and sometimes impractical to naïvely apply single-prover RA techniques to each device in a potentially large swarm that might cover a large physical area. Second, swarm RA needs to take into account topology discovery, key management and routing. This can be further complicated by mobility and device heterogeneity, in terms of computing and communication resources.

A recently proposed scheme, called SEDA: Scalable Embedded Device Attestation [2], represents the first step towards practical swarm RA. SEDA builds upon existing hybrid SMART and TrustLite [6] techniques. It combines them with a flooding-like protocol that propagates attestation requests and gathers corresponding replies. Despite its viability as a paper design, SEDA is not a practical technique, for several reasons. First, it is under-specified in terms of: (1) impact of swarm RA on the underlying hardware and security architecture, (2) overall attestation timeout determination for the verifier, and (3) selection criteria for the initiator device(s) that start(s) the attestation process in order to construct a spanning tree. Second, SEDA has some gratuitous (unnecessary) features, such as the use of public key cryptography, which are unjustified by the assumed attack model. Third, it is unclear whether SEDA handles mobility. This is an important issue: some swarm settings are static in nature, while others involve mobility and dynamic topologies.

Finally, SEDA does not capture or specify the exact quality of the overall attestation outcome and thus provides no means to compare security guarantees of various swarm RA techniques. We believe that it is important to define a qualitative (and whenever possible, quantitative) measure for swarm RA, i.e., *Quality of Swarm Attestation* (QoSA). This measure should accurately reflect verifier's information requirements and should also allow us to compare multiple swarm RA techniques.

Contributions: In order to bring swarm attestation closer to practice, after defining the notion of QoSA, we design and evaluate two practical swarm RA protocols (***LISA***α and ***LISA***s) that narrow the gap between paper-design techniques such as SEDA and realistic performance assessment and practical deployment. We also carefully investigate their impact on the underlying security architecture. Performance of proposed protocols is assessed using the open-source Common Open Research Emulator (CORE) [1].

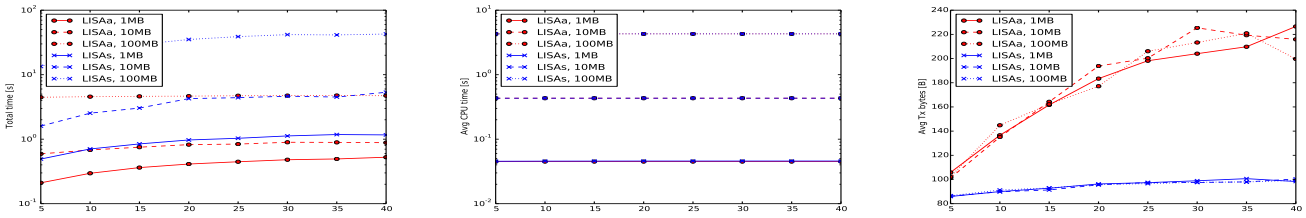
II. Quality of Swarm Attestation (QoSA)

The main goal of swarm RA is to verify collective integrity of the swarm, i.e., all devices at once. However, in some settings, e.g., when a swarm covers a large physical area, the granularity of a simple binary outcome is not enough. Instead, it might be more useful to learn which devices are potentially infected, so that quick action can be taken to fix them. By the same token, it could be also useful to learn the topology. To this end, we introduce a notion that tries to capture the information provided by swarm RA, called *Quality of Swarm Attestation (QoSA)*. It also enables comparing multiple swarm attestation protocols. We consider the following types of QoSA:

- *Binary QoSA (B-QoSA):* a single bit indicating success or failure of attestation of the entire swarm.
- *List QoSA (L-QoSA):* a list of identifiers (e.g., link-layer and/or network-layer addresses) of devices that have successfully attested.
- *Intermediate QoSA (I-QoSA):* information that falls between B-QoSA and L-QoSA, e.g., a count of successfully attested devices.
- *Full QoSA (F-QoSA):* a list of attested devices along with their connectivity, i.e., swarm topology.

Note that, in a single-prover setting which applies to most prior attestation literature, QoSA is irrelevant, since $\mathcal{V}$rf communicates directly with one $\mathcal{P}rv$, and there is no additional

[*]Work conducted while at HRL Laboratories



(a) Total time vs # of devices. (b) Time per device vs # of devices. (c) Transmitted data per device vs # of devices

Figure 1: Experimental Results for ***LISA***α and ***LISA***s

information beyond the attestation result itself. In contrast, in a multi-prover setting, QoSA is both natural and useful. It can be tailored to the specific application's needs.

III. New Swarm RA Protocols

Architectural minimality is a key goal of this work; hence, our proposed protocols require minimal hardware support. Specifically, we assume that each device adheres to the SMART architecture [5], augmented with $\mathcal{V}$rf authentication (aka DoS mitigation) features identified in [3]. We refer to this combination as ***SMART+***.

A. Asynchronous Version: LISAα

LISAα stands for: **L**ightweight **S**warm **A**ttestation, **a**synchronous version. Its goal is to provide efficient swarm RA while incurring minimal changes over ***SMART+***. In a very intuitive approach, $\mathcal{V}$rf, relying strictly on ***SMART+***, runs an individual attestation protocol directly with each swarm device. This would require no extra support in terms of software or hardware features. Nonetheless, this approach does not scale, since it requires $\mathcal{V}$rf to either: (1) attest each device in sequence, which can be very time-consuming, or (2) broadcast to all devices and maintain state for each, while waiting for replies. This scalability issue motivates device collaboration for propagating attestation requests and reports. ***LISA***α adopts this approach and involves very low computational overhead, while being resistant to computational denial-of-service (DoS) attacks. Each device keeps track of its parent upon receiving an attestation request. Once a request is processed and propagated, a device acts independently and asynchronously, relying on another device only for forwarding reports.

B. Synchronous Version: LISAs

The main idea in ***LISA***s is to let devices authenticate and attest each other. When one device is attested by another, only the identifier of the former needs to be securely forwarded to $\mathcal{V}$rf, instead of the entire report. This translates into considerable bandwidth savings and lower $\mathcal{V}$rf workload. Also, reports can be aggregated, which decreases the number of packets sent and received. It also allows more flexibility in terms of QoSA: from B-QoSA to F-QoSA. Finally, malformed or fake reports are detected in the network and not propagated to $\mathcal{V}$rf, as in ***LISA***α. However, these benefits are traded off for increased protocol (and code) complexity, as described below.

LISAs's main distinctive feature is that each device waits for all of its children's reports before submitting its own. This makes the protocol synchronous. Each device keeps track of its parent and children during an attestation session. Once a request is processed and propagated, a device waits for each child to submit its attestation report. Then, a device verifies each report, aggregates a list of children and all their descendants, attests itself, and finally sends the aggregated report to its parent.

IV. Experimental Assessment

We implemented ***LISA***α and ***LISA***s in Python, and assessed their performance by emulating device swarms using the open-source Common Open Research Emulator (CORE) [1]. Total time varies significantly between ***LISA***α and ***LISA***s. In ***LISA***s, nodes spend a lot of time waiting for external input, without computing anything. In these results, the factor varies between 2 (for 1MB) to 8 (for 100MB). This time is also heavily influenced by the size of the attested memory. Finally, total attestation time depends (roughly logarithmically) on number of nodes in a swarm, since nodes are explored in a tree fashion. Bandwidth usage is, as expected, higher in ***LISA***α than in ***LISA***s. The exact difference depends on n, ranging from negligible (5 nodes) to 3 (40 nodes). This only represents payloads size. Nodes in ***LISA***α also send more packets, compared to only 3 in ***LISA***s. Bandwidth usage is roughly linear in terms of number of nodes.

V. Conclusions and Future Work

This paper brings swarm RA closer to reality by designing two simple and practical protocols: ***LISA***α and ***LISA***s. To analyze and compare multiple protocols we introduced a new metric, called Quality of Swarm Attestation (QoSA) which captures the type of information offered by a swarm RA protocol. We believe that QoSA is of independent interest. Issues for future work include: (i) formally proving security for swarm protocols, and (ii) trial deployment of proposed protocols on device swarms. Our full paper can be found at [4].

References

[1] J. Ahrenholz, "Comparison of core network emulation platforms."

[2] N. Asokan, F. Brasser, A. Ibrahim, A.-R. Sadeghi, M. Schunter, G. Tsudik, and C. Wachsmann, "SEDA: Scalable embedded device attestation," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 964–975.

[3] F. Brasser, K. B. Rasmussen, A.-R. Sadeghi, G. Tsudik, I. Martinovic, K. B. Rasmussen, M. Roeschlin, G. Tsudik, G. Revadigar, C. Javali *et al.*, "Remote attestation for low-end embedded devices: the provers perspective," in *Design Automation Conference (DAC)*, 2016.

[4] X. Carpent, K. ElDefrawy, N. Rattanavipanon, and G. Tsudik, "Lightweigh swarm attestation: a tale of two lisa-s," in *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, 2017.

[5] K. Eldefrawy, G. Tsudik, A. Francillon, and D. Perito, "SMART: Secure and minimal architecture for (establishing dynamic) root of trust." in *NDSS*, vol. 12, 2012, pp. 1–15.

[6] P. Koeberl, S. Schulz, A.-R. Sadeghi, and V. Varadharajan, "TrustLite: A security architecture for tiny embedded devices," in *Proceedings of the Ninth European Conference on Computer Systems*. ACM, 2014, p. 10.