Poster: Enhancing Remote Healthiness Attestation for Constrained IoT Devices

Yihao Jia, Bingyang Liu, Weiyu Jiang, Bo Wu, Chuang Wang
Huawei Technologies Co. Ltd., Beijing, China

***Abstract*—The Internet of Things (IoT), which has been rapidly implemented in the smart home, city, and industry, keeps shaping the way we live. However, the constrained resource of IoT leads to a constant vulnerability for its' resident network and the whole Internet. To mitigate potential threats, a complementary method – Device Identifier Composition Engine (DICE) – is introduced to enable remote healthiness attestation for IoT devices. Although DICE narrows the gap between security necessity and the constrained resource of IoT, a replay attack is still possible to circumvent the method. In this paper, an enhanced DICE+ is proposed to address the weakness. Compared to the original DICE, DICE+ improves DICE with dynamic attestation evidence (other than static evidence in standard DICE), and thus alleviates the replay attack. Based on the evaluation, DICE+ enhances the standard DICE in three aspects *simultaneosly*: (*i*) Replay attack resilience; (*ii*) Extreme lightweight overhead; (*iii*) Fine-grained firmware attestation. According to the chip specification from our product line, a *ca.* 60% size reduction of the chip security-related area is expectable if such the method applied along with a pure symmetric-cryptography tech-set.**

I. Introduction

Although security is no doubt the most significant portion of Internet of Things (IoT), lots of IoT devices in reality lack appropriate defense ability for its naturally constrained resource [1]. To remain in a low energy consumption level, mechanisms, including those for security purposes, have to be simplified or simply cut. Given that being immune to attacks is completely impractical for IoT, the 'remote attestation' technology provides an alternative remediation by enabling networks to dynamically evaluate the device status, *e.g.*, whether their functions have been tampered. Figure 1 depicts a brief description of the universal remote attestation.

To balance the requirement of remote attestation and the constrained resource of IoT, a tailored framework – Device Identifier Composition Engine (DICE) – is proposed [2]. Particularly, there are two disparate specifications of DICE: symmetric cryptography based DICE and asymmetric cryptography based DICE. Although both specifications are considered to be an appropriate practice in balancing complexity and energy consumption, they all face challenges: the symmetric one fails in providing fine-grained firmware attestation, while the asymmetric one is less friendly to the highly constrained devices. Most importantly, both mechanisms fail to resist replay attacks due to the static attestation evidence.

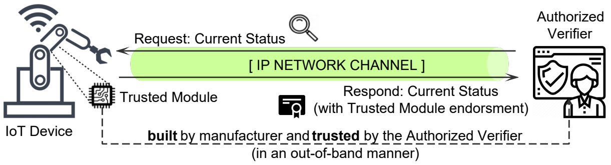
Authorized
Verifier
Request: Current Status
[IP NETWORK CHANNEL]
Trusted Module
Respond: Current Status
(with Trusted Module endorsment)
IoT Device
built by manufacturer and trusted by the Authorized Verifier
(in an out-of-band manner)

Fig. 1. Remote Attestation: whenever a current status check in needed, a challenge will be requested. Once challenged, a trusted module inside the device will collect the information needed, and respond with its' endorsement.

In this paper, we enhance DICE with DICE+. Specifically, DICE+ uses a new algorithm with a counter under symmetric cryptography, and such changes update the static attestation evidence to the dynamic, and thus alleviate the replay attack. Based on the evaluation, DICE+ enhances the standard DICE in three aspects *simultaneosly*: (*i*) replay attack resilience; (*ii*) extreme lightweight overhead; (*iii*) fine-grained firmware attestation. According to our product line, a *ca.* 60% size of chip security-related area is potentially reducible if DICE+ along with a pure symmetric-cryptography tech-set applied.

II. Design

Since DICE+ in this paper is built on the standard DICE, we follow the official route and expound below.

1) Objective: In a nutshell, DICE enables remote evaluation for the comprehensive healthiness of IoT devices by identifying the integrity and the version of the IoT firmware. Considering that the IoT is usually resources constrained, the logical composition of it is relatively simple. For this reason, the functions, usually, are directly built in the firmware without independent applications upon it. As a result, the key to the remote healthiness attestation for constrained devices is to identify the firmware version and firmware integrity.

2) Principle: Specifically, the firmware, as depicted later in Figure 2, is devised in a hierarchical multi-layer, and the higher the layer is, the more functions and attacking surface there will be. Thus, the function of the root layer is devised to be extremely simple and bug-free if elaborately programmed. Under such philosophy, all layers higher than the root are supposed to be hack-feasible.

3) Prerequisite: Based on the principle, a DICE module is placed in the root (layer -1) and stores a Unique Device Secret (*UDS*) value. As the name implied, the *UDS*, which installed during manufacture, has high value and *must* be delivered and stored in a secure manner. In theory, the DICE module process as follows to keep *UDS* safe and make itself be unconditionally trusted: (*i*) be the only layer that can access the *UDS*; (*ii*) immediately shut itself off right after 'layer 0' start.

978-1-7281-6992-7/20/$31.00 ©2020 IEEE

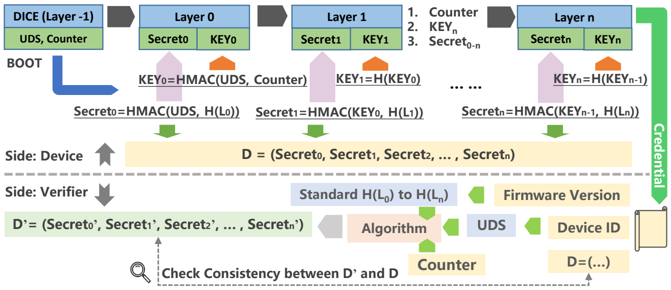

Fig. 2. The process of remote healthiness attestation in DICE+

4) Process: The entire process is divided into three parts. First, each DICE built-in device stores a *UDS* (in layer -1), a monotonically increasing counter *CNT* (in layer -1, and increase every time the device boot), a firmware version *VER* (in layer n). These values are all bind with a unique device *ID*, *e.g.*, a unique device sequence number. Before putting into use, *UDS*, *CNT*, and the correlated *ID* should be shared with the remote verifier in an out-of-band manner and used as the endorsement later.

Second, the device firmware is booted through a layer-by-layer sequence. To formulate the process, we depict as follows. Mathematically, let L_i depict the static firmware code of layer i, and *H( )* and *HMAC( )* refer to the function of 'hash' and 'keyed-hash message authentication code', respectively. During the boot of each layer, *e.g.*, layer i, a KEY_{i+1} and a $Secret_{i+1}$, as expressed below, would be calculated.

$$Secret_{i+1} = \begin{cases} HMAC(H(L_{i+1}), KEY_i) & i >= 0 \\ HMAC(H(L_{i+1}), UDS) & i = -1 \end{cases}$$

$$KEY_{i+1} = \begin{cases} H(KEY_i) & i >= 0 \\ HMAC(UDS, CNT) & i = -1 \end{cases}$$

Then layer i boot the next layer $i + 1$, and deliver KEY_{i+1}, *CNT*, and $D = (Secret_0, Secret_1 ..., Secret_{i+1})$ to it until all layers are booted. At the end of the boot, the firmware will hold the following credentials: *ID*, *VER*, *CNT*, and $D = (Secret_0, Secret_1, ..., Secret_n)$.

Finally, whenever being challenged for the attestation, the device will securely respond with these credentials whether in an explicit or implicit way. On the verifier side, *UDS* will be retrieved according to *ID*, while the standard hash values for each layer, which acquired from the firmware publisher in advance, will be retrieved according to *VER*. Based on *UDS*, standard hash values, and the same computation algorithm, the verifier output the standard $D^{'} = (Secret^{'}_1, Secret^{'}_2, ..., Secret^{'}_n)$. Thus, the firmware integrity can then be confirmed if D equals to $D^{'}$, and the specific impaired layer i can be pointed out where $Secret_i$ differs from $Secret^{'}_i$.

5) Feasibility Analysis: The mechanism above supports the remote attestation for the device healthiness on two aspects: (*i*) Any impermissible/malicious code embedment incurs a firmware change, and thus results in a completely different hash output; (*ii*) For devices attested to have integral firmware, the firmware version may also imply a potential threat. If not timely updated, the devices with outdated firmware could lead to an explicit attack surface. Even for firmware with the latest version, any vulnerability published by a white-hat hacker can still expose the devices in grave danger.

III. Preliminary Evaluation

DICE+ enhances DICE in three aspects *simultaneously*:

- **Replay attack resilience** via a monotonically increasing counter. For standard DICE specification, the evidence, mainly refer to D, is constant and always valid until a firmware update is applied. By introducing an increasing counter, the evidence changes once the device reboots. Thus, such mechanism enable networks to identify a replay attack by discerning the outdated evidence, behind which imply a hacked device and extensive threats.
- **Extreme light weight overhead** via pure symmetric cryptography. Compared to the standard DICE with asymmetric cryptography, energy consumptions for methods with pure symmetric cryptography can be reduced by 1000x on average [3]. According to the chip specification from our product line, the size of the chip security-related area can be reduced *ca.* 60% if DICE+ is applied along with a pure symmetric-cryptography tech-set.
- **Fine-grained firmware attestation** via the new algorithm. For the standard DICE with symmetric cryptography, the evidence provides the integrity check for the firmware as a whole, missing positioning capability for the specific tampered layer. Thanks to KEY_i, the value of $Secret_i$ no longer correlates to $Secret_{i-1}$ as in standard DICE, thus each $Secret_i$ becomes independent and represent the integrity for each layer of the firmware.

IV. Practice

According to the technical context, the DICE-related mechanisms is a research branch of the "secure boot", and thus be classified to the *system* security, other than the *network*. For constrained devices that cannot validate firmware integrity themselves, a remote assistant is required. Hence, it is just the assistant that gets the "network" involved and overlaps security of the *system* and the *network*. However, the remote attestation has not been paid enough attention in network security area until the concept of "zero trust network" has been brought to the forefront. According to the "zero trust", all network interactions will evolve into a "trust but verify" mode, and such mode exactly and *naturally* accord with the process – remote attestation. Given that the dominant position of IoT in the foreseeable future, remote attestation will certainly play a decisive role in networks with constrained devices.

References

[1] O. Garcia-Morchon, S. Kumar, and M. Sethi, "Internet of Things (IoT) Security: State of the Art and Challenges," RFC 8576, Apr. 2019.

[2] DICE Working Group, Trusted Computing Group. [Online]. Available: https://trustedcomputinggroup.org/work-groups/dice-architectures

[3] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha, "Analyzing the energy consumption of security protocols," in *Proceedings of the international symposium on Low power electronics and design*, 2003.