

An IoT Device-trusted Remote Attestation Framework

Kyeong Tae Kim

Information Security Research Division

Electronics Telecommunications Research Institute

161 Gajeong-dong, Yuseong-gu, Daejeon, 305-700, KOREA
 ktkim@etri.re.kr

Jeong-Nyeo Kim

Information Security Research Division

Electronics Telecommunications Research Institute

161 Gajeong-dong, Yuseong-gu, Daejeon, 305-700, KOREA
 jnkim@etri.re.kr

Jae Deok Lim

Information Security Research Division

Electronics Telecommunications Research Institute

161 Gajeong-dong, Yuseong-gu, Daejeon, 305-700, KOREA
 jdscol92@etri.re.kr

Abstract—Remote attestation allows the verifier to determine the level of trust for the system's health. We propose a framework for verifying the trustworthiness of the IoT device using remote attestation. The framework provides remote device trust verification and a trust management service for a large group of devices to prevent remote control and zombization due to firmware spoofing. We show how the framework performs remote attestation for object forgery attacks under various types of heterogeneous TEE (SW) and TPM (HW) IoT platforms.

Index Terms—Remote Attestation, Security, IoT

I. INTRODUCTION

Recently, there was a crash with IoT devices easily leaking personal information due to a malicious DDoS attack. CCTV cameras (around 150,000 devices) were hacked to become part of a bot and attack a French web hosting company [1]. If we periodically monitor the status of IoT devices such as CCTV cameras (the internal situation if the device has been infected with malicious code by a malicious hacker), then this type of damage can be prevented.

Integrity refers to the characteristics of the data on the platform that is kept without being exposed to intentional modification or destruction. Remote attestation uses a protocol to verify its integrity by sending it to the verifier, which determines the level of trust for the system health of the IoT device.

The concept of a remote attestation framework is shown in Fig. 1. In the remote attestation (integrity verification), the remote attestation server verifies whether the integrity of the files is compromised with the trust verification agent operating in the remote attestation device. This protocol is performed remotely, i.e., over a network. The goal of the protocol is to allow an honest (not compromised) device to create an authentication token, that convinces server that the measurement is in some well-defined (expected) state. Whereas, if the device has been compromised and its state has been modified, the authentication token must reflect this. The measurement is to collect evidence about the state

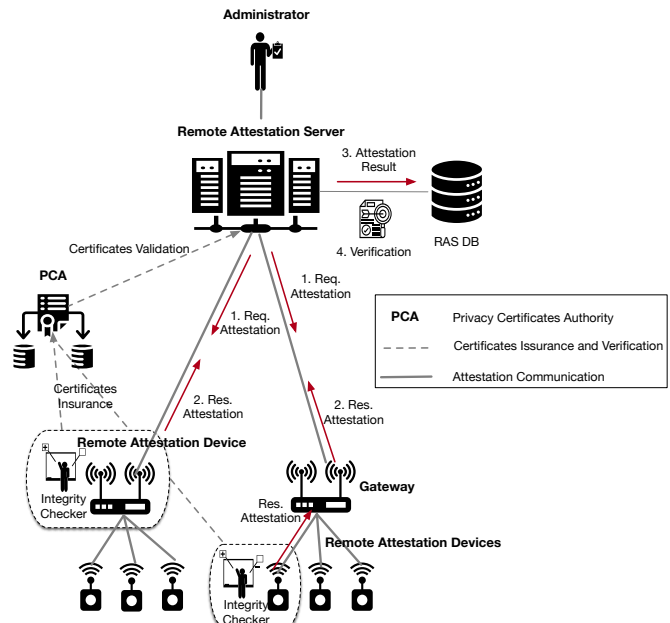


Fig. 1. The Concept Diagram of A Device-trusted Remote Attestation Framework.

through direct and local observation of it. The measurement mechanism specifies what parts of the run-time environment should measure, when and how securely maintain the measurements. This integrity challenge mechanism is to allow authorised server to retrieve measurement list, verify freshness and completeness [12]. PCA (Private Certificate Authority) creates (issues) certificates for use within a private network or within an organization. The request to periodically perform attestation on device and its response (result) are maintained by RAS (Remote Attestation System) DB. Remote attestation framework based on Root of Trust, a source that can always be trusted within a cryptographic system, assumes either a shared secret between server and device (i.e., a symmetric key k), or

public key cryptography techniques (e.g., traditional public key cryptography).

In this paper, we propose an IoT device-trusted remote attestation framework, which includes a device-trusted remote verification and trust management service for a large group of devices to prevent remote control and zombieization due to firmware forgery. The scope of the application of trusted remote attestation targets networking IoT devices. It remotely verify whether individual objects (files) in the IoT device are damaged in real-time.

The framework is compatible with various lightweight IoT devices: Raspberry Pi based on TPM [9] in Raspbian OS, NXP Saber SD board based on OP-TEE [10] in Wayland OS (general environment OS) and OP-TEE OS (trusted environment OS), and Raspberry Pi based on OPEN-TEE [11] in Ubuntu (general environment OS) and OPEN-TEE OS (trusted environment OS).

The contribution of this paper is twofold: real-time monitoring of massive malware infections of IoT devices and heterogeneous support of the TEE (Trusted Execution Environment) platform. In real-time monitoring of malware infection, the framework diagnoses large-scale IoT device integrity in real-time and restores damaged objects. It detects damaged objects by remote verification and restores damaged objects using secure storage in the TEE environment. It also supports cluster-based remote trust verification for scalability.

In case of support of heterogeneous TEE platforms, different TEE platforms are provided according to the characteristics of the various IoT devices such as ARM TrustZone and Hypervisor. Various remote verification elements (TPM, OP-TEE, OPEN-TEE, etc.) can be expanded. These various device platforms can interact through the cloud service as a common remote trust management framework.

The rest of the paper is organized as follows. We detail the proposed device-trusted remote attestation framework in detail in section 2. In section 3, we verify the effectiveness of our framework through the experiment. Related works are discussed in section 4. Finally, we conclude this paper and suggest future work in section 5.

II. AN IOT DEVICE-TRUSTED REMOTE ATTESTATION FRAMEWORK

Device-trusted remote attestation framework is composed of the RAS (Remote Attestation System) RA (Remote Attestation) server subsystem, which is the subject of remote attestation for each device, and the RAS RA device subsystem, which generates device verification information and communicates with the RAS RA server subsystem.

This section describes the roles of the RAS RA device subsystem and the RAS RA server subsystem that are subsystems constituting the RAS system.

A. The RAS RA Device Subsystem

The RAS RA device subsystem verifies the integrity of itself and delivers verification information to the RAS RA server. It contains an application that executes remote attestation using

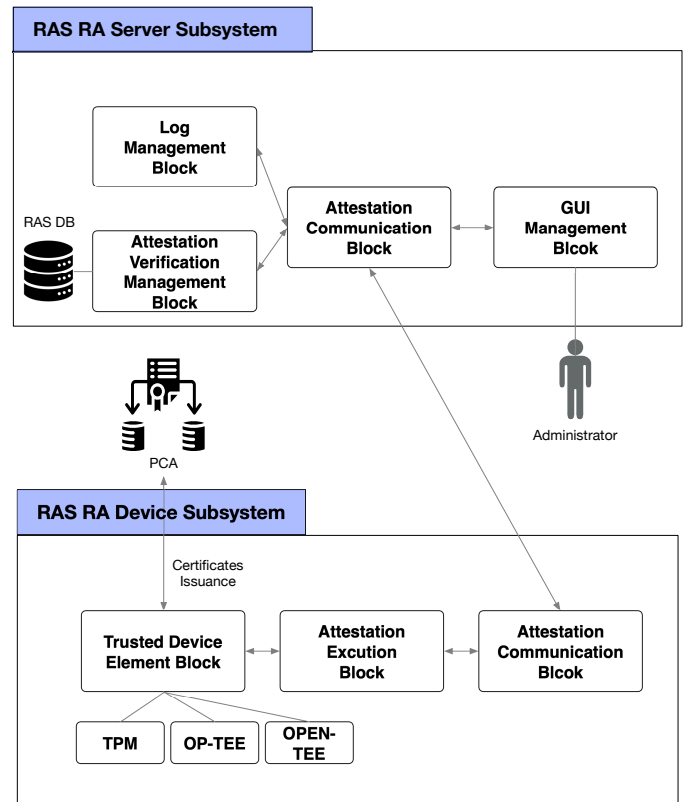


Fig. 2. The Architecture of A Device-trusted Remote Verification Framework.

the trusted device element and registers itself with the RAS RA server during initial boot. It periodically receives remote verification requests and performs integrity verification. The main functions are as follows.

- To generate individual or integrated verification information for devices through remote attestation agents.
- The integrated verification information is transmitted as a signed verification value using the device security element before being sent to the RAS RA server subsystem.

As seen in Fig.2, the main block of the RAS RA device subsystem is the trusted device element block. It is an HW/SW module installed in IoT devices to provide trust in the integrity verification value. It consists of TPM, OP-TEE, and OPEN-TEE. The trusted device element block detects tampered objects with remote attestation in large-scale environments and recovers tampered objects using TEE local secure storage. It can recover all or individual objects for detected corrupted objects.

The attestation communication block is to register the device to the remote verification server. It also receives and processes remote verification requests from a remote verification server, and transmits measured device integrity verification value as a response.

The attestation execution block is to obtain integrity verification values for files existing in the monitoring directory stored in the configuration file. It also calculates a hash value for device integrity using a trusted device elementbased

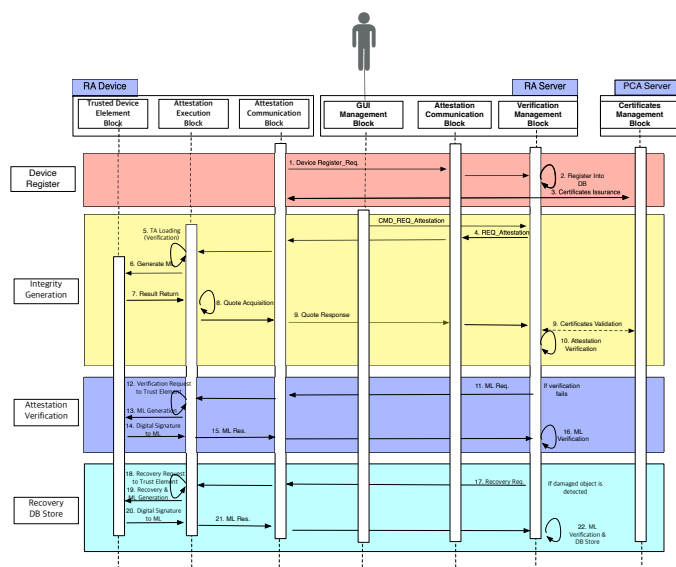


Fig. 3. The Interfaces Between Subsystem

interface. Finally, it creates a measurement list and chained hash value (using SHA256 Hash function)

B. The RAS RA Server Subsystem

The RAS RA sever subsystem is a subject requesting to perform remote verification by communicating with the RA device subsystem, it checks the integrity using the remote attestation and manages the result. The RAS RA server subsystem is a management application for servers that registers and manages multiple applications. It receives the RAS RA device subsystem registration request, registers it in the DB, and periodically requests remote verification for the registered RAS RA device subsystem. In addition, it receives the remote verification request from the GUI web server and delivers it to the RAS RA device subsystem. All contents are managed by DB. The main functions are as follows.

- Verification of individual/integrated verification information transmitted from the device.
- Generation of individual verification information: Hash value for each object (program, etc.) to be verified.
- Generation of integrated verification information: Hash value chain for individual verification information.
- Management of verification information and verification results for each device.
- To provide device validation status via a web interface.

In Fig.2, the RAS RA server subsystem communicates with the RAS RA device subsystem and requests to perform remote verification, it checks integrity using the remote attestation and manages the result. It consists of a log management block, attestation verification management block, attestation communication block, and GUI management block.

The log management block maintains log table history.

The attestation verification management block is to registers in DB by receiving a device registration request from RAS RA

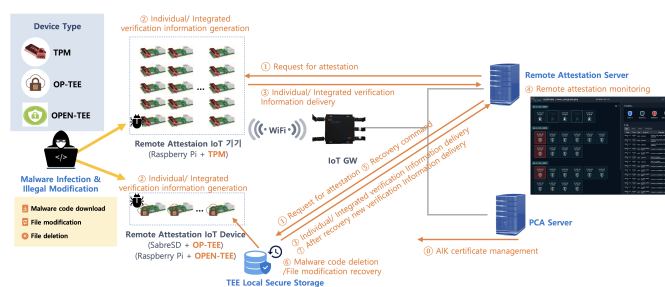


Fig. 4. The Attack Scenario.

device subsystem and stores integrity information of multiple devices.

The attestation communication block sends remote verification requests and response results to the remote verification device. The request to periodically perform attestation on devices is registered in DB.

The GUI management block is a graphical web interface for remote verification management that shows the user the status of remote verification of IoT devices. It provides device remote verification status information based on DB for remote trust verification management.

C. The diagram of the interaction between subsystems

Fig.3 illustrates a schematic diagram of the interaction between subsystems. When the system boots up, in the device registration phases, the RAS RA device subsystem proceeds with device registration during the initial connection with the RAS RA server subsystem (step 1). The individual and integrated verification information of the device along with the device configuration information is registered in the RAS RA server subsystem (step 2).

In the device verification request and integrated verification delivery phase, the RAS RA server subsystem periodically requests verification information with a random nonce from the device for device verification (step 4). The device receiving the request calculates the individual verification information for each object to be verified for integrity and then calculates the integrated verification information based on the individual verification information (steps 5 and 6). The calculated integrated verification information is stored in the trusted device element block through the API that can utilize TEE (Trusted Execution Environment) to ensure that it is calculated normally and not corrupted by the device and then loaded with the 'Quote' API. The 'Quote' is delivered to the RAS RA server subsystem as a certificate of the trusted device element block (steps 7 and 8).

In the device platform remote verification phase, the RAS RA server subsystem verifies whether the device is damaged by comparing the integrated verification information received from the device with the integrated verification information stored when the device is registered (step 10). If the integrated verification information does not match during the comparison process, it means that there is a change in the objects to be verified, so request verification information for each object to the

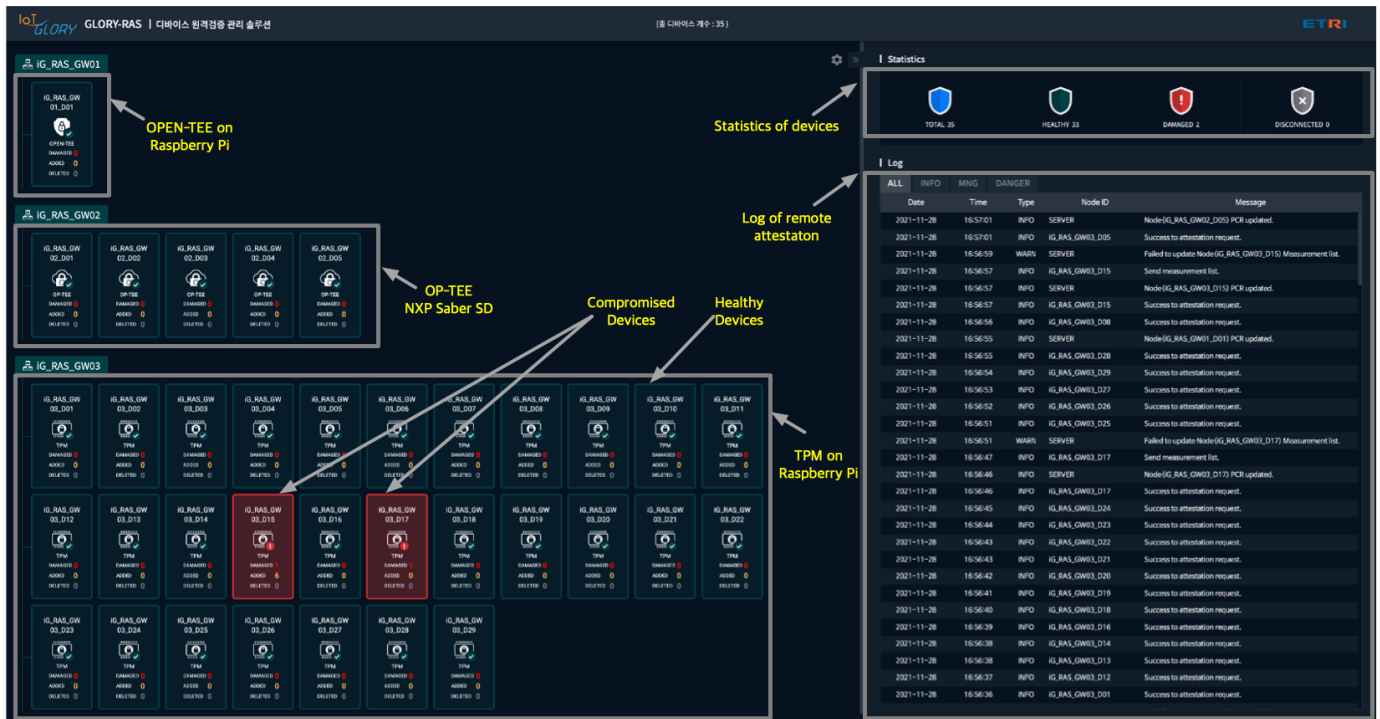


Fig. 5. The Remote Attestation Monitoring Server.

RAS RA device subsystem is delivered (step 11). The device transmits the calculated individual verification information list to the RAS RA server subsystem (step 12). The RAS RA server subsystem checks individual verification information and reports to the manager what object has changed (steps 13, 14, 15, and 16).

In the object recovery and saving verification results phases, the RAS RA server subsystem compares the individual verification information received from the device with the individual verification information stored when the device is registered, checks whether the object is damaged, and requests recovery from the trusted device element (steps 17 and 18). After performing full or individual recovery on the detected damaged object, remote verification information is created and delivered to the RAS RA server subsystem (steps 19, 20, 21, and 22).

III. EVALUATION

In this section, we show how the framework performs remote attestation for object forgery attacks. As seen in Fig. 4, the testbed is constructed for the verification of individual/integrated verification information delivered from 3 types of devices: TPM (30 units), OP-TEE (5 units), and OPEN-TEE (1 unit).

In TPM, secure cryptographic services (encryption, decryption, hash, etc.) are implemented in HW. A lightweight remote verification protocol supports large-scale IoT environments. The remote verification is based on global standards (TCG v1.2)

In OP-TEE(SW), secure cryptographic services are implemented in SW. It is based on global standards (GlobalPlatform

TEE API). An execution environment guarantees authentication and integrity and increases security stability by isolating user executable code and data access.

In OPEN-TEE(SW), secure cryptographic services are implemented in SW. It is based on global standards (GlobalPlatform TEE API) It supports a "virtual TEE" that complies with the GP (GlobalPlatform) TEE specification using a hypervisor. Trusted applications developed using a virtual TEE can be compiled and run against any target that conforms to the specification.

Fig. 4 shows the attack scenario for 3 types of devices. The system checks the results of remote attestation after adding/deleting/modifying files for remote verification of TPM-based devices. It also monitors remote verification after performing the malicious code infection process. When the malware infection on OP-TEE and OPEN-TEE-based devices through a malicious code infection emulation script occurs, malicious code installation detection and recovery results are confirmed through remote attestation. Here, malware code is added to /opt/ra_directory by running the malware infection script in OP-TEE and OPEN-TEE-based devices. Individual verification information is hashed value for each object (program, etc.). On the other hand, integrated verification information is a chained hash value for individual verification information.

Fig. 5 shows the GUI for the status of the remote attestation monitoring server. According to the scenario, it shows the result of each device remote verification through the web interface. As seen in Fig. 6, the system recovers (or deletes) damaged objects using secure storage existing in TA (Trusted

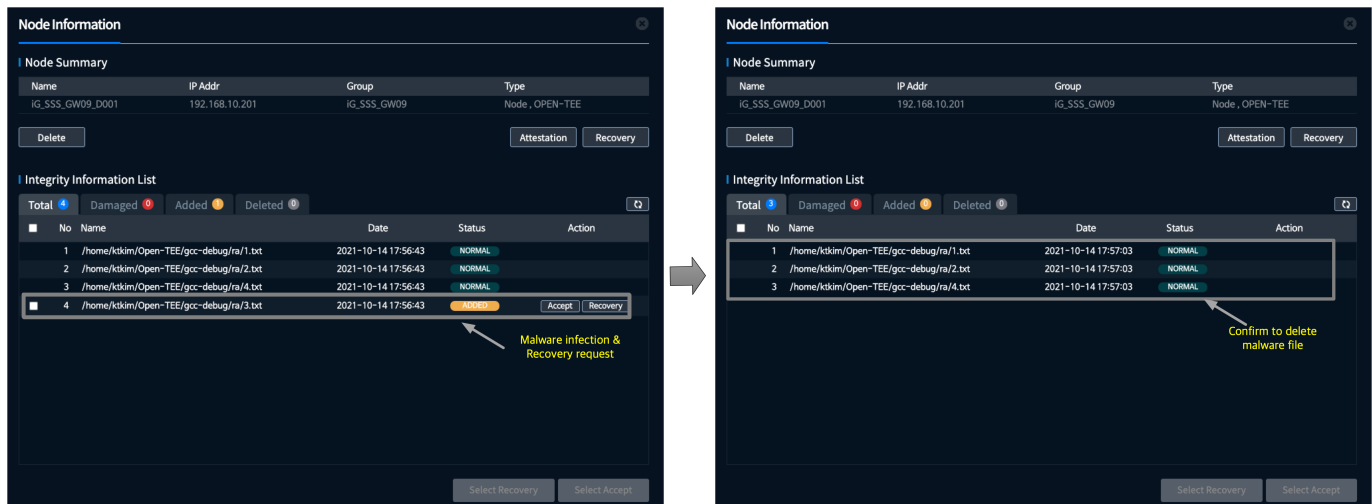


Fig. 6. The Malware Infection and Recovery.

Application). As a result of remote verification, it is confirmed that the integrity is damaged due to malicious code download. Object backup and recovery are performed under secure storage in TA.

IV. RELATED WORKS

For large-scale remote attestation, several works have been done to allow the verifier to "collect" the network status and optionally obtain the individual status of each device, which effectively improves the scalability of the RA protocol.

SEDA [2] is the first collective attestation solution based on the hybrid model. It uses a spanning tree for routing construction. The limitation of this, however, is that each device maintains a fixed set of neighbors for the duration of the operation. The settings are also stationary in the offline phase.

SANA [3] is an attestation protocol based on asymmetric primitives using the Optimistic Aggregate Signature (OAS) scheme.

A tale of Two LISA-s [4] offers Quality of Swarm Attestation (QoSA): Binary; Count; List; Full. This improves SEDA in terms of scalability and flexibility against physical enemies. The limitation, however, is that full communication between the devices is required during the attestation process.

RADIS [5] improves the state of the art by tackling runtime attacks.

SARA [6] aims to attest distributed IoT services running on IoT devices (Publish/Subscribe). However, the limitation is that it requires full connectivity among devices. Also, the aggregate signature scheme puts a heavy load on low-end devices in terms of computing power.

DARPA [7] provides absence detection to detect physical attacks using capturing missing heartbeat absent devices. Devices periodically run an absence detection protocol with their neighbors, recording present devices in a secure log. These devices generate "heartbeat" messages with a time stamp

signed with their secret key and share them with immediate neighbors.

SCAPI [8] improves DARPA. It mitigates against physical attacks. SCAPI security is based on the assumption that an attacker cannot capture a device without deactivating it for an extended period of time.

V. CONCLUSION

We propose a device-trusted remote attestation framework that supports various heterogeneous IoT devices: TPM (HW) based IoT device (RPi platform), OP-TEE (SW) based IoT device (ARM platform), and OP-TEE(SW) based IoT device (Hypervisor platform). The framework is optimized for hyper-connected, large-scale, and lightweight device environments.

In the framework, the secure trust element provides ARM TrustZone based TA and hypervisor-based virtual TA. Both are implemented based on global standards (GlobalPlatform TEE API). Real-time detection of malware infection and its recovery can be archived by integrity verification using remote attestation. As soon as tampered objects are detected, the framework recovers them using TEE secure storage. The framework can recover any or all of the objects from the detected damaged objects.

Malware detection and device health verification are security-critical infrastructure services. It can be used as a security monitoring and control framework for security threats in large-scale IoT infrastructures, such as enterprises, institutions, and national defense.

ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2018-0-00231, Development of context adaptive security autonomous enforcement technology to prevent spread of IoT infrastructure attacks.

REFERENCES

- [1] <https://www.infosecurity-magazine.com/news/major-french-web-hosting-company-hacked/>
- [2] N. Asokan, F. Brasser, A. Ibrahim, A.-R. Sadeghi, M. Schunter, G. Tsudik, and C. Wachsmann, "SEDA: Scalable embedded device attestation," in ACM SIGSAC Conference on Computer and Communications Security (CCS), 2015.
- [3] Moreno Ambrosin, Mauro Conti, Ahmad Ibrahim, Gregory Neven, Ahmad-Reza Sadeghi, and Matthias Schunter, "SANA: Secure and scalable aggregate network attestation.", In ACM SIGSAC Conference on Computer and Communications Security, 2016.
- [4] X. Carpent, K. ElDefrawy, N. Rattanavipanon, and G. Tsudik, "Lightweight Swarm Attestation: a Tale of Two LISA-s," in Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, ser. ASIACCS '17. ACM, 2017
- [5] M. Conti, E. Dushku, and L. V. Mancini, "RADIS: Remote Attestation of Distributed IoT Services," in 2019 Sixth International Conference on Software Defined Systems, ser. SDS '19. IEEE, 2019, pp. 25–32.
- [6] E. Dushku, M. M. Rabbani, M. Conti, L. V. Mancini, and S. Ranise, "Sara: Secure asynchronous remote attestation for iot systems," IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3123–3136, 2020.
- [7] Ahmad Ibrahim, Ahmad-Reza Sadeghi, Gene Tsudik, Shaza Zeitouni, "DARPA: Device Attestation Resilient to Physical Attacks", in WISEC, 2016.
- [8] Kohnhäuser, Florian and Büscher, Niklas and Gabmeyer, Sebastian and Katzenbeisser, Stefan, "Scapi: a scalable attestation protocol to detect software and physical attacks," in Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks, 2017
- [9] <https://trustedcomputinggroup.org>
- [10] <https://www.op-tee.org>
- [11] <https://open-tee.github.io>
- [12] G. Coker, J. Guttman, P. Loscocco, A. Herzog, J. Millen, B. O'Hanlon, J. Ramsdell, A. Segall, J. Sheehy, and B. Sniffen, "Principles of Remote Attestation," in International Journal of Information Security, 10(2):63 81, 2011.



secure OS, system and network security.

Jeong Nyeo Kim received her M.S. degree and Ph.D. in Computer Engineering from Chungnam National University, Rep. of Korea, in 2000 and 2004, respectively. She studied at computer science from the University of California, Irvine, USA in 2005. Since 1988, she has been a principal member of engineering staff at the Electronics and Telecommunications Research Institute (ETRI), where she is currently working as a Project Leader of the Network System Security Research Section. Her research interests include IoT security, mobile security,



Kyeong Tae Kim received the B.S. degree in Computer Engineering from Kangwon National University, Korea in 2004 and the M.S. degrees in Information and Communications from Gwangju Institute of Science and Technology, Korea in 2006, respectively. Since 2006, he has been a research member of Electronics and Telecommunications Research Institute (ETRI). His research interests include Network Security and Wireless Communication.



Jae Deok Lim received his M.S. degree in Electronic Engineering from Kyungbook National University, Korea in 2001 and his Ph.D. M.S. degree in Computer Engineering from Chungnam National University, Korea in 2013. He is currently a senior research member at Information Security Research Division in Electronics and Telecommunications Research Institute(ETRI). His research interests include IoT security, access control, secure operating system.