Poster: Secure Logging Infrastructure Employing Heterogeneous Trusted Execution Environments

Seungho Lee*, Wonsuk Choi*, Hyo Jin Jo†, Dong Hoon Lee*
*Graduate School of Information Security, Korea University, Seoul, South Korea
Email: {seculee,beb0396,donghlee}@korea.ac.kr
†School of Software, Hallym University, Chuncheon, South Korea, Email: hyojinjo86@gmail.com

***Abstract*—In this present hyper-connected era, interconnected heterogeneous devices send logs to a distributed cloud storage in real-time, and transmitted logs must be accessible only to authorized users. These logs are referred to as fine-grained access-controlled logs. For the fine-grained access control to logs, we propose a secure logging infrastructure employing heterogeneous trusted execution environments that allows a logging infrastructure administrator to operate the system robustly without significant risk of key exposure, malicious activity, or abnormal behavior. In our method, embedded devices based on ARM architecture encrypt their own logs using a key-policy attribute-based encryption scheme in ARM TrustZone, which are then sent to the distributed cloud storage. After this, service providers based on Intel architecture download the logs from the distributed cloud storage and decrypt them with a decryption key that is delivered through a secure channel established in Intel software guard extensions. At this point, access control to logs is possible since the decryption key includes a policy related to log access permissions.**

I. Introduction and Motivation

Numerous heterogeneous devices are connected by one network in this hyper-connected era. Each device generates logs when an event occurs inside the system and sends this log to a remote server or distributed cloud storage when necessary. Since logs are human-readable messages that are platform-independent, service providers can use them to detect abnormal behavior or provide suitable services to customers by analyzing logs without additional pre-processing. However, existing logging systems do have limitations in terms of only providing reliable logs to authorized service providers.

In [2], Karande et al. provided a method that generated reliable logs by encrypting the data in a hardware-based isolated execution environment called Enclave, which was provided by Intel software guard extensions (SGX). However, it proved impossible to apply this proposed method to ARM-based systems, which accounts for the majority of embedded devices. Besides, since this method did not provide fine-grained access control to logs, it would have been difficult for many authorized service providers to utilize the logs. To resolve this problem, Lee et al. proposed a secure logging method suitable for ARM-based embedded devices[4]. The authors applied a key-derivation scheme based on a hash-chain and generated digital signatures for the blocks containing numerous logs. In [3], Lee et al. extends the previous work so as to support fine-grained access control to logs by applying enhanced key policy attribute-based encryption (KP-ABE) called *e-ABE* that improves the key encapsulation mechanism of KP-ABE, which reduces computational overhead and provides performance tuning parameters for operating on various platforms. The authors solved the key revocation issue from the original attribute-based encryption by keeping the decryption key confined to ARM TrustZone and restricting direct user access. However, most service providers use a high-performance computing environment based on Intel architecture. Thus, decrypting logs in ARM TrustZone is quite inefficient in this environment. Therefore, in this paper, we propose a more practical and efficient method that employs high-performance Intel SGX for log decryption, while satisfying the same security requirements proposed in [3]. In addition, the proposed method, which uses heterogeneous trusted execution environment (TEE), guarantees reliable operation among system operators, embedded devices, and service providers in a secure logging infrastructure. The contributions of our method are as follows:

- Proposing a secure logging infrastructure that employs heterogeneous TEEs.
- Providing ARM TrustZone vs. Intel SGX performance comparison results and performance profiling results.
- Sharing background knowledge and major differences in commercial TEE solutions.

II. ARM TrustZone vs. Intel SGX

The representative commercial TEE solutions are ARM TrustZone and Intel SGX. ARM TrustZone is primarily used for embedded devices and Intel SGX is used for high-performance desktops or servers.

TABLE I: Comparison of ARM TrustZone with Intel SGX

Feature	ARM TrustZone	Intel SGX
Business domain	Embedded system	Desktop, Server
Performance	Mid or Low	High
Chain of trust	Secure boot chain	N/A
Attestation	N/A	Local/Remote
Trusted object	Trusted application (TA)	Trusted part of app.
Transition path	Monitor software in EL3	Hardware-based Ring-3 (user)
Call interface	Secure monitor call	Edge routines (ECall/OCall)
Key provisioning	Manufacturing	Remote provisions by Intel
Dependency	Vendor	Commercial license

TABLE I presents major security function differences. Among these, the items that Intel SGX can quickly execute trusted code in Ring-3 without switching the execution mode and service providers commercially licensed with Intel that can operate trusted solutions in Enclave are attracted. In addition, the administrator can remotely discern by the remote attestation feature in Intel SGX if the released binary is running normally in the Enclave without any indication of abnormal behavior. Owing to these advantages, we were assured that Intel SGX would be a suitable solution for service providers since the operator of the logging infrastructure can efficiently share and manage sensitive data or keys through a secure channel.

III. System Architecture

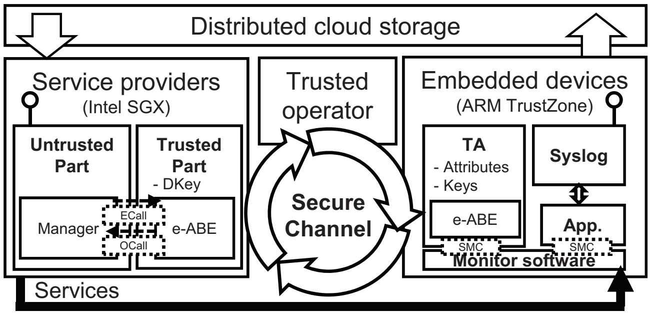


Fig. 1: The system architecture of a logging infrastructure that uses heterogeneous trusted execution environments

Fig. 1 shows the architecture of a logging infrastructure. The components of a logging infrastructure are composed of trusted operators who manage keys and policies, embedded devices that generate logs, and service providers that provide customized services by analyzing logs stored on a cloud storage.

Embedded devices use *e-ABE* for embedded systems in [3] to encrypt logs with specific attributes, and these logs are subsequently sent to cloud storage. Accordingly, service providers are only able to access attribute-based encrypted logs that correspond to a policy contained in the decryption key (DKey) issued by a trusted operator. In this paper, we employ Intel SGX for more practical and efficient operation execution than the method using only ARM TrustZone, as proposed by Lee et al. in [3]. The untrusted part downloads logs from cloud storage and calls ECall to enter the Enclave, and then the decryption module of the trusted part decrypts the logs without key exposure and returns to the caller. If the trusted part needs to use specific functions in the untrusted part, it enables to request through calling OCall. The interface of ECall and OCall is automatically generated by Edger8r tool, which translates enclave definition language syntax to header and source files.

In addition, the operator can prevent running a maliciously manipulated binary across the whole system with the assistance of heterogeneous TEEs because ARM TrustZone only loads a verified binary through a secure boot-chain scheme, and Intel SGX ensures that only signed codes are loaded in Enclave with a commercial license.

IV. Implementation and Evaluation

In order to develop Intel SGX, both a supported motherboard and an SGX-enabled processor are necessary. Additionally, for ARM TrustZone, a development board supporting all TEE software packages is required. We have implemented a service provider on Gigabyte B360HD3, Intel Core i7 9770KF, and the Raspberry Pi3 B+ embedded system, and the crypto library has been ported for each TEE with the Celia library[5] modified from CP-ABE[1].

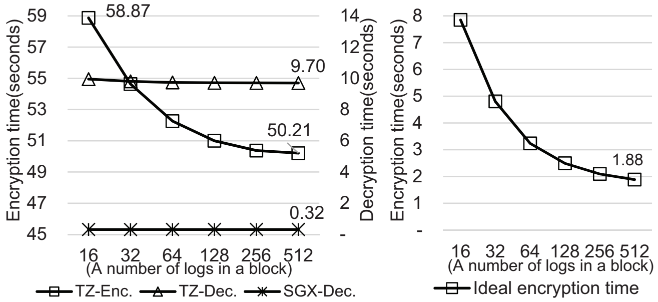


Fig. 2: Results of performance measurement

The left side of Fig. 2 compares decryption time on ARM TrustZone (TZ-Dec.) with Intel SGX (SGX-Dec.), while the right side of the figure shows ideal encryption time on ARM TrustZone. Decrypting 1,000 logs sized 512bytes based on ARM TrustZone took approximately 9.7 seconds, but, under the same conditions, it only took Intel SGX approximately 0.32 seconds, which is about 30 times faster. In addition, after performance profiling, we found that most of the encryption overhead in ARM TrustZone occurs when encrypted logs are transferred from TA to Syslog. As a result, our proposed scheme will be able to encrypt 1,000 logs within 1.8 seconds, indicating that it can theoretically handle 532 logs per second (LPS). Furthermore, an increase in the number of logs in a block can also improve performance without dramatically affecting overhead.

V. Conclusion and Discussion

In this paper, we proposed a secure logging infrastructure employing Intel SGX to provide a highly secure and efficient environment, as an extension of the method in [3]. However, in order to be practically applicable in the industry, further research into assistant tools that could operate on a logging infrastructure more conveniently is needed, and performance improvement is necessary to maximize the number of logs that can be handled per second.

References

[1] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," [Online; accessed 10-Jan-2020]. [Online]. Available: http://acsc.cs.utexas.edu/cpabe

[2] V. Karande, E. Bauman, Z. Lin, and L. Khan, "SGX-Log: securing system logs with SGX," in *Proc. 2017 AsiaCCS*. ACM, 2017, pp. 19–30.

[3] S. Lee, W. Choi, H. J. Jo, H. Kim, J. H. Park, and D. H. Lee, "A dependable and secure logging method with fine-grained access control on ARM TrustZone," *IEEE Internet of Things journal*, [Under Review, Submitted in 24 Nov., 2019].

[4] S. Lee, W. Choi, H. J. Jo, and D. H. Lee, "How to securely record logs based on arm trustzone," in *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, 2019, pp. 664–666.

[5] Y. Zheng, "Privacy-preserving personal health record system using attribute-based encryption," 2011.

NDSS Symposium 2020

Secure Logging Infrastructure Employing Heterogeneous Trusted Execution Environments

Seungho Lee[1], Wonsuk Choi[1], Hyo Jin Jo[2] and Dong Hoon Lee[1]
[1]Graduate School of Information Security, Korea University [2]School of Software, Hallym University

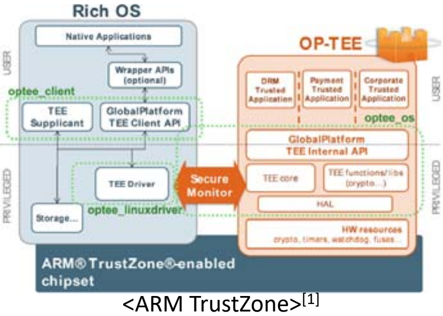
MOTIVATION

- Logs are human-readable messages that are platform-independent.
- Logs are used to detect abnormal behavior, analyze root causality, and aid digital forensics.
- Numerous heterogeneous devices are connected by one network in our hyper-connected era.

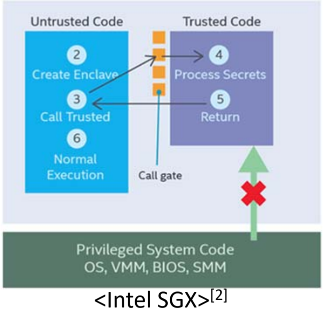
"How can we guarantee logs are trustworthy??"

Making a secure logging infrastructure suitable for our hyper-connected era.

TEE SOLUTIONS


Rich OS
USER
PRIVILEGED
Native Applications
Wrapper APIs (optional)
optee_client
TEE Supplicant
GlobalPlatform TEE Client API
TEE Driver
Storage...
optee_linuxdriver
Secure Monitor
OP-TEE
DRM Trusted Application
Payment Trusted Application
Corporate Trusted Application
optee_os
GlobalPlatform TEE Internal API
TEE core
TEE functions/libs (crypto...)
HAL
HW resources
crypto, timers, watchdog, fuses...
ARM® TrustZone®-enabled chipset

<ARM TrustZone>[1]



<Intel SGX>[2]

[1]https://www.linaro.org/blog/lcu14-107-op-tee-armv8/
[2]https://software.intel.com/en-us/articles/intel-software-guard-extensions-tutorial-part-1-foundation

ARM TrustZone vs. Intel SGX

Feature	ARM TrustZone	Intel SGX
Business domain	Embedded system	PC and Server
Performance	Mid or Low	High
Secure storage	Secure file system	Sealing / Unsealing
Chain of Trust	Secure boot chain	N/A
Trusted object	Trusted Application	Trusted part of an App.
Transition path	Monitor software in exception level 3 (EL3)	Hardware-based Ring-3 (user mode)
Call interface (Inst.)	API (SMC)	Edge routines (ECall/OCall)
Key provisioning	Key injection in manufacturing	Remotely provisioned by Intel
Attestation	N/A	Local/Remote
Dependency	Vendor	Commercial license
Standards	GlobalPlatform TEE[3]	Intel
Open source	ARM Trusted Firmware[4] OP-TEE[5]	Linux-SGX[6]
Supplements	Bootloader, Secure OS, Driver, Daemon, SDK	SDK

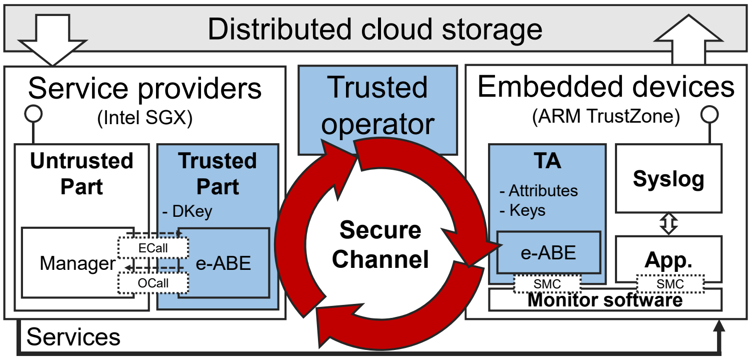
[3]https://globalplatform.org/technical-committees/trusted-execution-environment-tee-committee
[4]https://github.com/ARM-software/arm-trusted-firmware [5]https://www.op-tee.org
[6]https://github.com/intel/linux-sgx

CONTRIBUTIONS

- Proposing a secure logging infrastructure that employs heterogeneous TEEs.
- Providing ARM TrustZone vs. Intel SGX performance comparison results and performance profiling results.
- Sharing background knowledge and major differences in commercial TEE solutions.

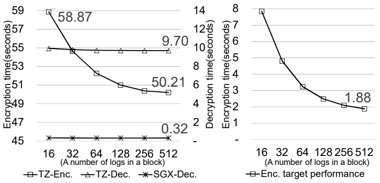
SYSTEM DESIGN

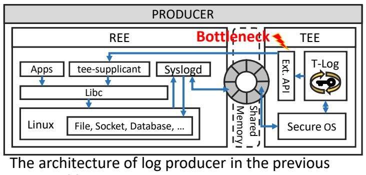
Architecture



IMPLEMENTATION & EVALUATION

Category	ARM TrustZone	Intel SGX
Processor	Broadcom BCM2837 64bit Quad Core 1.2GHz	Intel Core i7 9770KF 64bit Octa Core 3.6GHz
Board	Raspberry Pi3 B+	Gigabyte B360HD3





The architecture of log producer in the previous research[7]

- Under identical conditions, Intel SGX took approximately 0.32 seconds, which is about 30 times faster than ARM TrustZone.
- If we resolve the bottleneck in [7], our method will be able to encrypt 1,000 logs within 1.8 seconds and can theoretically handle a minimum of 532 logs per second.

[7]S. Lee et al. "A dependable and secure logging method with fine-grained access control on ARM TrustZone "IEEE Internet of Things journal , [Under Review, Submitted in 24 Nov, 2019].

FUTURE WORK AND DISCUSSION

- Optimizing performance to process the maximum number of logs per second.
- Interacting with big data, blockchain, and machine learning technologies.
- Researching on visualization, provenance inference, and other related areas.