Security SoC Architecture with Hardware-Based Pre-Authentication

Won Bae Kong[1], Pil Joo Choi and Dong Kyue Kim[a]
Department of Electronic Engineering, Hanyang University
E-mail : [1]wbgong@hanyang.ac.kr

***Abstract* – Edge Devices with limited power and processing performance need the help of hardware-based security solutions in order to provide sufficient security services. The hardware-based security solutions have been proposed to separate hardware resources into the secure area and the normal area and determine whether accessible according to the secure mode of the processor. The solutions determine secure mode based on the importance of the application or to let the user decide. However, this makes it possible for unauthorized users to access the secure area when the device is stolen or replicated. To solve this problem, we propose a hardware-based pre-authentication protocol which determines if the edge device is a safe situation. The proposed pre-authentication protocol includes all the processes the chip producing, issuing, and using. SoC with the Core-A processor and pre-authentication module was implemented as a hardware chip, and it was confirmed that Core-A enters secure mode after succeeding in the pre-authentication protocol.**

***Keywords*—Authentication protocol, Hardware security, Security SoC**

I. INTRODUCTION

With the recent advances in smartphones and IoT technology, tasks that were previously only available on PCs have become possible in edge devices. These tasks have expanded to include financial, healthcare, and transportation, which can affect users' personal information, property, and safety. As hackers who have been performing attacks in the PC environment also attack users using security vulnerabilities of edge devices, it is essential to apply security solution to edge devices. However, it is difficult to apply the heavy security software used in PCs to edge devices which have limited power and processing performance. As a result, a variety of hardware-based security solutions [1-10] have been proposed to reduce the load on processing performance and available in low-power environments.

As one of the hardware-based security solutions, it has been proposed for separating the hardware processing environment according to the application. Intel's software guard extensions (SGX) [1] and ARM's Trustzone [2] are solutions that separate hardware resources into normal area and secure area, and access area according to the secure mode of the processor. Intel's SGX allows the user to determine the processor's secure mode through software, and ARM's Trustzone determines the processor's secure mode based on the importance of the application. However, these solutions do not distinguish between whether the current edge device is in a safe or dangerous situation, it can be accessible to the secure area even when the edge device is stolen or replicated and can pose a security threat.

We propose a hardware-based pre-authentication protocol to determine if the edge device is in a safe situation. Only processors on edge devices authenticated through the pre-authentication protocol can enter secure mode. It can prevent the processor from entering the secure mode when the device is lost, stolen, or replicated. The authentication protocol meets the following security requirements:

- Object authentication: If the protocol succeeds, it must verify the identity of the object participating in it.
- Key exchange: If the protocol succeeds, participants in the protocol should be able to share a secure session key.
- Confidentiality: While the protocol is proceeding, sensitive information contained in messages should not be able to be identified by an attacker.
- Integrity and non-repudiation: Sensitive information in the protocol must not be tampered with by the attacker, and messages that are approved by each participant should not be denied subsequently.
- Prevent reuse attacks: If an attacker saves some of the messages from a performed protocol and then reuses them later, participants should be able to recognize them.
- Preventing man-in-the-middle attacks: When an attacker attempts a man-in-the-middle attack, there should be no additional information or permissions obtained by the attacker compared to normal circumstances.

This paper consists of: Chapter 2 describes the subject of the protocol and the pre-authentication protocol process, and Chapter 3 analysis the security of the authentication protocol. Chapter 4 describes the hardware-based pre-authentication protocol and the SoC structure that was implemented and concluded in chapter 5.

a. Corresponding author; dqkim@hanyang.ar.kr

Manuscript Received Sep. 02, 2019, Revised Sep. 09, 2019, Accepted Sep. 20, 2019

II. PRE-AUTHENTICATION PROTOCOL

In this section, we show the subject of the protocol and describe protocol process details.

A. Protocol Subject

The subject spearheading the protocol is a chip, manufacturer, issuer, instrument, and trusted service manager (TSM). The relation between each subject is shown in Fig. 1. The description of each subject is as follows.

Fig. 1. The relation between subjects of pre-authentication protocol

a. Chip

The security chip mounted inside the device has a PUF-based key that enables identification of the chip. The PUF-based key is the private key (Pv_{chip}) of the public-key cryptography and the symmetric key (Sm_{chip}) of the symmetric-key cryptography. The chip includes a hardware module that performs a cryptographic algorithm (public-key cryptography, symmetric-key cryptography, hash function, random number generation) and must be able to store private information securely and perform knowledge-based certification.

b. Manufacturer (MF)

A chip manufacturer is responsible for everything involved in chip manufacturing, including physical chip manufacturing and the installation of the necessary software. It gives the chip a serial number (SN), and the connection between the chip and the manufacturer is considered as a trust interval.

c. Issuer & Issuing Machine (IM)

The issuer is the subject issuing the chip to the user, such as the bank. The issuer makes the chip issued through the issuing machine, and the connection between the chip and the issuing machine is considered as a trust interval. The chip that has been issued becomes available.

d. Trusted Service Manager (TSM)

As a subject that ensures the reliability of the chip, it manages the chip's ID and public key and issues a certificate of the chip's public key (Pb_{chip}). TSM has its private key (Pv_{TSM}) and public key (Pb_{TSM}) to use the public-key cryptography. The communication between the TSM and the issuer is considered as a trust interval, and the communication between the TSM and the chip can be dangerous.

e. Device

The device, including the chip, support the operating system and hardware for i/o interface with a user, communication with TSM. The device assumes that it is not safe because it is possible to attack, such as hacking, and therefore, the communication channel (communication between device and TSM) is not safe.

B. Protocol Process

The pre-authentication process can be divided into three main stages: The Manufacturing stage of manufacturing security chips in the factory, the Issuing stage of issuing security chips to the user, and the Using stage of using security chips to operate the application. Devices equipped with a security chip can activate the processor's secure mode permission after performing the pre-authentication at a power-on or periodically.

a. Manufacturing

In the manufacturing process, the manufacturer gives SN to the chip and manages the SN list given to the chip. The chip has an SN and PUF-based key (Pv_{chip}, Sm_{chip}) inside. The manufacturer delivers the finished chip to the issuer, such as the bank, and delivers the SN list of the manufactured chips to the TSM.

b. Issuing

Issuing is the process of exchanging information between the chip and the TSM. At this time, the communication between the chip and TSM is made through the IM of the issuer. IM is regarded as a reliable device, and communication between chips and issuing devices, and issuing devices and TSM are also regarded as a trust interval. The issuing process can be divided into three primary steps:

1. The chip sends its SN to the TSM and generates a Pb_{chip} for Pv_{chip}.
2. Chips and TSM exchange each other's public keys. TSM sends its Pb_{TSM} to the chip, and the chip sends its Pb_{chip}.
3. TSM generates a certificate ($Cert_{chip}$) for Pb_{chip} and sends to the chip.

After the issuing is terminated, the chip has its SN, Sm_{chip} , Pv_{chip} , Pb_{chip} , $Cert_{chip}$, and Pb_{TSM} . TSM manages the SN and Pb_{chip} as a list. The chip that is issued is delivered to the user and ported to the device.

c. Using

The process of being certified by TSM before the device's processor enters secure mode so that users can safely use

services such as mobile banking. The communications between chip and TSM are non-trust interval and use security protocol since it through the user, the app, and the device. The authentication process for the chip and the device includes forming a security channel session for further information exchange with the TSM. The process of use is divided into six steps:

1. Comparing the device authentication information (HK_{device}) stored on the device to the authentication information that was encrypted existing in the chip, the chip authenticates the user and the device.
2. The chip and TSM certify each other through the SN, the other party's public key, and the certificate. Depending on the public-key cryptography used, detailed steps of the signature and authentication algorithm [11] may vary.
3. Chip and TSM generate a one-time share key using public-key cryptography and a random number generator. Depending on the public-key cryptography, the detailed key sharing algorithm [12] may be different. A one-time share key is used as a session key for a session.
4. Using the session key, symmetric-key cryptography, and message authentication code (MAC) algorithm, chip, and TSM perform secure communication.
5. Before the session ends, send the chip's signature of the message used in the session to TSM.
6. Send the hash value (HK_{device}) of the session key used in this session to the device and store the value encrypted (EHK_{device}) by symmetric-key cryptography and Sm_{chip} to the chip. Hashed session key value is used in the following authentication as authentication information for the device.

After authentication to the chip and device through the Using stage, the processor gains permission to enter secure mode. After the Using stage end, the chip has its SN, Sm_{chip}, Pv_{chip}, Pb_{chip}, $Cert_{chip}$, Pb_{TSM}, and EHK_{device}. TSM has an SN-Pb_{chip} list, and the device has a HK_{device}.

III. SECURITY ANALYSIS OF PROTOCOL

This chapter analyzes the security features provided by the proposed protocol. MF and IM, which are only involved in the manufacturing and issuing, are always assumed to be safe communication. We analyze the security of chips, TSM, and device objects and describes security features that satisfy.

A. Object authentication

Pre-authentication's Using stage provides mutual authentication and key exchange between chip and TSM. The chip and TSM can verify the identity of the other party by using the public key stored in the issuing step, confirming that the other party is the authorized owner of the private key associated with the public key. The identity of the device can be confirmed by HK_{device}, which indicates 'the same device as the device that was connected to the previous protocol.'.

It can be assumed that the TSM is attacked and critical information about the chip is leaked. Even so, the leaked SN and public key of the chip does not make the attacker can disguise it as a legitimate chip.

B. Key exchange

After performing the pre-authentication's Using stage, the chip and TSM are securely shared session keys by public-key cryptography [12].

C. Confidentiality

Session keys are securely shared through public-key cryptography. Messages encrypted with session keys are securely protected between chip-TSM and cannot be verified by an eavesdropper.

D. Integrity and Non-repudiation

The MAC of messages generated by session keys is securely protected between chip-TSM and causes errors when tampering in the intermediate. The signature of the public-key cryptography ensures integrity and non-repudiation.

E. Prevent reuse attacks

Reuse attacks are impossible because we use a random number generator to generate shared keys in public-key cryptography. The generated shared key is used only as a one-time session key.

F. Prevent man-in-the-middle attacks

Since the chip and TSM have the other party's public key in advance, all the messages sent are guaranteed their validity by signature, the attacker cannot deceive and intervene in the identity.

IV. IMPLEMENTATION AND CHIP VERIFICATION

A. SoC Implementation

We designed the security SoC with the hardware-based pre-authentication protocol. The structure of the entire security SoC is shown in Fig. 2. The hardware-based pre-authentication protocol module consists of a logic part that controls the protocol process, via PUF cell [13] for chip recognition and key generation, and cryptographic modules that operate cryptography algorithms. The other part of SoC was configured using the Core-A processor [14] and static random-access memory (SRAM) to determine whether entering the processor's secure mode depending on pre-authentication protocol success.

The cryptographic modules consist of the symmetric-key cryptography algorithm AES [15] and SEED [16], the public-key cryptography algorithm ECC [17], hash algorithm SHA1[18] and SHA2 [18], and true random number generator (TRNG) [19] for generating random numbers. For protocol control and parameter storage, non-volatile (NV) memory and RAM are used, which both are SRAM. The universal asynchronous receiver/transmitter (UART) protocol which is serial communication, is used as an external communication interface.

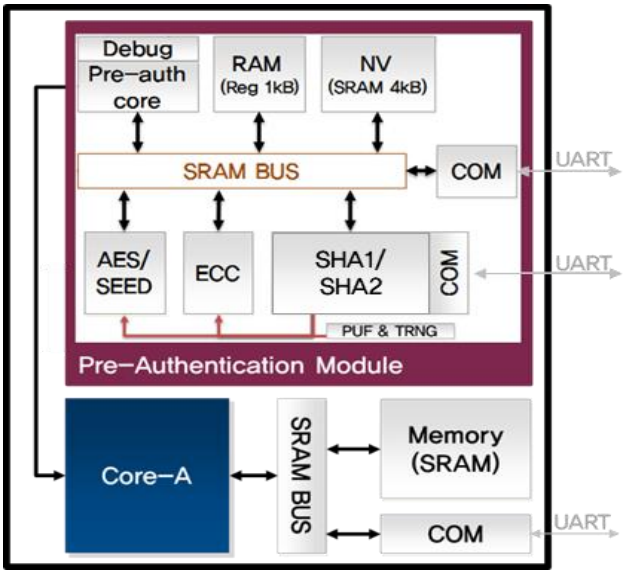


Fig. 2. The structure of implemented SoC

We implemented the security SoC using the Samsung 60 nm complementary metal-oxide-semiconductor (CMOS) application-specific integrated circuit (ASIC) technology library. The security SoC was implemented at 20Mhz clock frequency. The post-synthesis results of the pre-authentication module are shown in Table 1. The gate count of the whole pre-authentication module and each submodule are presented separately.

TABLE I. Gate Count of Pre-Authentication Protocol

Module	Gate Count
Pre-Authentication Total	353,662
AES	33,897
SEED	14,700
ECC	139,034
SHA1	10,572
SHA2	15,194
TRNG	1,267
NV	1,409
RAM	86,513
UART	1,153 x 2

B. Chip verification

We used the test board to verify the implemented chips. The chip was operated at 40Mhz, and the data for verification was exchanged through UART communication with PC. The test board shown in Fig. 3 is connected to the PC.

We made a software program that performs the operation of the manufacturers, issuers, devices, and TSM to verify that the implemented chip performs the pre-authentication protocol correctly. It was confirmed that the chip performs the pre-authentication protocol correctly, and it was confirmed that the processor could enter the secure mode only if the authentication is successful. The pre-authentication protocol is performed through the software program shown in Fig.4.

Fig. 3. Test board for verify chip operation

Fig. 4. *The software program for verify pre-authentication protocol*

V. CONCLUSION

We proposed a way to protect the device's secure world from danger situation where the device is stolen or replicated by checking before the processor entering secure mode. We proposed the pre-authentication protocol, which can determine whether the device is safe or not and analyzed the security of the proposed protocol. The protocol provides the object authentication of edge devices, safe key exchange, confidentiality and integrity of information, non-repudiation, and prevention of reuse attack and man-in-the-middle attacks. It was implemented in a chip to verify that the processor enters secure mode only in safety situations that pass the pre-authentication protocol.

The hardware-based pre-authentication protocol we propose can authenticate the edge device, but the legitimacy of the software installed within the device is unknown. If the application running on the device is tampered with, or if a malicious program is performed to monitor what the user enters the device, the user may still be exposed to security threats. Therefore, hardware-based security solutions such as SGX and TEE or software-based security solutions must be added for a completely secure edge device execution environment.

ACKNOWLEDGMENT

This work is supported by IDEC.

REFERENCES

[1] V. Costan, and D. Srinivas. "Intel SGX Explained." *IACR Cryptology ePrint Archive* 2016, 086: 1-118, 2016.

[2] ARM Ltd. TrustZone. Available online at: http://www.arm.com/products/processors/technologies/trustzone.php.

[3] J. Burke, J. McDonald, and T. Austin, "Architectural support for fast symmetric-key cryptography," *ACM SIGOPS Operating Systems Review*, 34.5: 178-189, 2000.

[4] S. Tillich, J. Großschädl, and A. Szekely, "An instruction set extension for fast and memory-efficient AES implementation," *IFIP International Conference on Communications and Multimedia Security*, Springer, Berlin, Heidelberg, p. 11-21, 2005.

[5] S. Tillich and J. Großschädl, "Instruction set extensions for efficient AES implementation on 32-bit processors," *International workshop on cryptographic hardware and embedded systems*, Springer, Berlin, Heidelberg, p. 270-284, 2006.

[6] Xilinx Corp., "CryptoBlaze: 8-Bit Security Microcontroller," Available online at: http://www.bdtic.com/download/Xilinx/xapp374.pdf.

[7] K. Akdemir, M. Dixon, W. Feghali, et al., "Breakthrough AES performance with intel AES new instructions," White paper, June 2010.

[8] S. O'Melia and A. J. Elbirt, "Enhancing the performance of symmetric-key cryptography via instruction set extensions," *IEEE transactions on very large scale integration (VLSI) systems*, vol. 18, no. 11, pp. 1505-1518, 2009.

[9] L. Wu, C. Weaver, and T. Austin, "CryptoManiac: a fast flexible architecture for secure communication," *Proceedings 28th Annual International Symposium on Computer Architecture*, Göteborg, Sweden, pp. 110–119, 2001.

[10] R. Buchty, N. Heintze, and D. Oliva, "Cryptonite–A programmable crypto processor architecture for high-bandwidth applications," *International Conference on Architecture of Computing Systems*. Springer, Berlin, Heidelberg, p. 184-198, 2004.

[11] Kerry, C. F and C. R, "Digital signature standard (DSS)," *Federal Information Processing Standards Publication 186-4*, July 2013.

[12] IEEE Std 1363a, "IEEE Standard Specifications for Public-Key Cryptography-Amendment 1: Additional Techniques," Mar 2004.

[13] B. D. Choi, T. W. Kim, M. K. Lee, K. S. Chung and D. K. Kim, "Integrated circuit design for physical unclonable function using differential amplifiers." *Analog integrated circuits and signal processing* 66.3: 467-474, 2011.

[14] J. H. KIM, D. H. You, K. S. Kwon, E. J. Bae, W. Son, and I. C. Park, "Design of high-performance 32-bit embedded processor." *2008 International SoC Design Conference*. IEEE, p. III-54-III-55, 2008.

[15] Standard, NIST-FIPS "Announcing the advanced encryption standard (AES)," *Federal Information Processing Standards Publication 197*, vol. 197, pp. 1-51, Nov 2001.

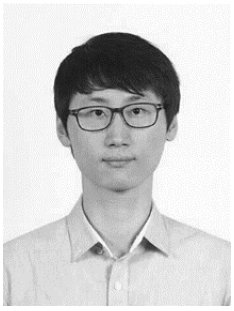
[16] H. J. Lee, S. J. Lee, J. H. Yoon, D. H. Cheon, J. I. Lee, "The SEED encryption algorithm." RFC 4269 (2005)

[17] V. S. Miller, "Use of elliptic curves in cryptography," *Conference on the Theory and Application of Cryptographic Techniques*, pp. 417-426, 1985.

[18] J. H. Burrows, *Secure hash standard*. Department of Commerce Washington DC, 1995.

[19] A. Rukhin, J. Soto, J. Nechvata, M. Smid and E. Barker, *A statistical test suite for random and pseudorandom number generators for cryptographic applications*. Booz-Allen and Hamilton Inc Mclean Va, 2001.

Won Bae Kong received the B.S. degrees in electronic engineering from Hanyang University, Seoul, South Korea, in 2015. He is currently a Ph.D candidate in electronics and computer engineering at Hanyang University. His research interests are in the areas of security SoC (System on Chip), secure crypto-processor, crypto-coprocessors, and information security systems.



Pil Joo Choi was born in Seoul, South Korea in 1982. He received the B.S., M.S., and Ph.D. degrees in electronic computer engineering from Hanyang University, Seoul, South Korea, in 2010, 2012, and 2018, respectively. He is currently a professor in Software Education Committee at Hanyang University. His research interests are in the areas of security SoC, crypto-coprocessors, and information security.

Dong Kyue Kim was born in Seoul, South Korea in 1968. He received the B.S., M.S. and Ph.D. degrees in computer engineering from Seoul National University in 1992, 1994, and 1999, respectively. From 1999 to 2005, he was an assistant professor in the Division of Computer Science and Engineering at Pusan National University. From 2006, he is a professor in the Department of Electronic Engineering at Hanyang University. His research interests are in the areas of security SoC, secure crypto-processor, crypto-coprocessors, and information security systems.