7-4 An Encrypted File Storage System with Hierarchical Levels of Sensitivity and its Applications

Lihua WANG, Takuya HAYASHI, Atsushi WASEDA, Ryo NOJIMA, and Shiho MORIAI

In this paper, we introduce a secure storage system named "PRINCESS" and an application, the PRINCESS-based automobile information sharing system that are developed utilizing a special proxy re-encryption technology that processed by NICT. Using PRINCESS, the files encrypted in accordance with the confidentiality levels can be shared among appointed users while remaining encrypted, so this system facilitates the potential for new services that require privacy data to be shared securely via cloud technology.

1 Introduction

In recent years, cloud storage services have become popular, but in many of the current cloud storage systems, data is uploaded to the storage server as-is, unencrypted. Thus, there is the danger of leaks in information of stored data, due to cyber-attacks, mistakes in operations by management companies, etc. Furthermore, depending on the members with whom the data has to be shared, it is desirable to be able to set the sharing policy (confidentiality level). Encryption of stored data is an effective means to resolve these problems. However, in conventional public key cryptography (for example, RSA encryption), a file that is encrypted with user A's public key can only be decrypted with user A's private key, so to share the encrypted file with multiple members, one must perform the encryption process one time for each person. On the other hand, when the file is encrypted using the same key shared among all members using symmetric key cryptography (for example AES), then the encryption process is needed only once for all persons to share, therefore it is possible to resolve the issue of public key cryptography. However, a different key must be used every time to ensure security, so the issues of key sharing and management remain. As one solution to these problems, the Security Fundamentals Laboratory developed PRINCESS (Proxy Re-encryption with INd-Cca security in Encrypted file Storage System) [1][2]. PRINCESS uses a NICT proprietary technology "IBPdr (IBE with functions of Proxy decryption and Proxy re-encryption)" [3][4], which is an encrypted file sharing system that considers the handling of user privacy and secret information. PRINCESS has the following characteristics.

- ID-based encryption (referred to hereinafter as IBE), for easy organizational management and migration from current storage systems
- Flexible information sharing inside and outside the organization, by 3 confidentiality level settings: "High," "Medium" and "Low"
- Flexible project and user management, by delegated authority revocation function

As a general encrypted file storage system, PRINCESS can be considered for various applications, such as back up of medical care data for quick recovery after disasters. As one example of an application, this paper describes an automobile information sharing system [5][6] developed based on PRINCESS. This system can flexibly share Controller Area Network (CAN) information of the vehicle, such as vehicle location fetched from GPS, vehicle speed and engine RPMs. With this, one can implement an automobile information sharing service that considers privacy protection.

2 PRINCESS

2.1 Cryptographic technologies used in PRINCESS

IBE: IBE is a kind of public key cryptography that features usage of a unique ID (for example, user's email address, etc.) as the public key of the user. For this reason, this encryption method is very convenient for users. Since the initial IBE [7] proposal by Shamir in 1984, various IBE schemes with provable security were proposed [8][9]. The IBE system is different from conventional public key cryptography systems where the private key for ID is issued by the user himself, in the fact that it is issued by a reliable

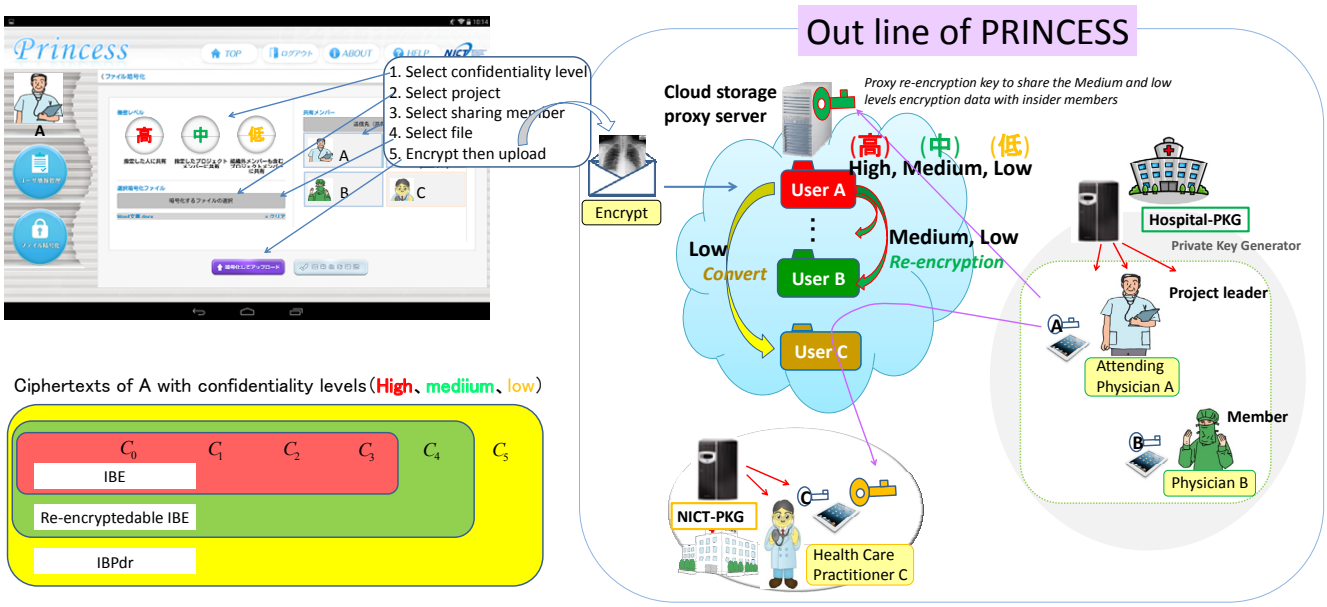


Fig. 1 PRINCESS overview

key generation center (Private Key Generator: PKG, for example, information management department of the organization).

Proxy decryption: "Proxy decryption" is an encryption method with a feature that enables a pre-designated party (a proxy) to decrypt ciphertext that conventionally only the legitimate receiver could decrypt. It was proposed as an encryption method with a proxy decryption function by Mambo et al. in 1997 [10]. In this case, the legitimate receiver passes the proxy a "proxy decryption key," rather than his/her own private key, and the proxy can use it to decrypt the ciphertext.

Proxy re-encryption: "Proxy re-encryption" was proposed by Blaze et al.[11]. With this function, user A, the legitimate receiver, passes a "re-encryption key" to a proxy server, which enables it to convert data encrypted for user A into data encrypted for user B, without decrypting the data first. Thereafter, B uses his "own private key" to decrypt the ciphertext after conversion. This re-encryption key has the characteristic that it can decrypt neither the ciphertext for A nor the text converted for B. Therefore, proxy re-encryption is an encryption technology that can share (with B) the data (of A) encrypted as-is (via a proxy server). As a method that extended proxy re-encryption, there is a method that can divide ciphertext into two confidentiality levels: ciphertext that can be re-encrypted, and ciphertext that cannot be re-encrypted [12][13].

We use a bilinear map to propose an ID-based cryptosystem, IBPdr, that can convert ciphertext created by an encryption method that has a proxy decryption function, which combines the three encryption techniques described above, into another user-addressed ciphertext by using proxy re-encryption. This method has the characteristic of dividing the ciphertexts into three confidentiality levels, so it enables more flexible data sharing [3][4].

Hybrid encryption: An encryption method that combines public key cryptography and symmetric key cryptography, which uses public key cryptography to deliver the symmetric key which is to be used to encrypt the data. In PRINCESS, data is encrypted using an AES symmetric key cryptosystem, and that AES key is encrypted in IBPdr, and saved along with the encrypted data file in the server.

2.2 PRINCESS overview and characteristics

In PRINCESS, encryption confidentiality levels are categorized into "High," "Medium" and "Low," depending on the scope of destinations for sharing. Ciphertext that can be shared only with specific members is defined as "High" level, ciphertext that can be shared within members inside the organization is defined as "Medium" level, and the ciphertext that can be shared with members including external associates is defined as "Low" level. The ciphertexts of "High," "Medium" and "Low" levels are each encrypted by IBE, re-encryption enabled IBE, and IBPdr, and are in a comprehensive structure as in Fig. 1 on the lower left. Ciphertext with a lower confidentiality level can be converted to a higher level confidentiality ciphertext, but the conversion from a higher level to a lower level is impossible.

This system is comprised of multiple organizations and a cloud storage proxy server, where each organization has a key generation center and users affiliated to this organization. It performs the following steps.

(1) System preparation: Each key generation center (for example, Hospital-PKG, NICT-PKG in Fig. 1) creates public parameters for its internal information system, issues private keys for employees in the organization (SK_{id}), and creates re-encryption key seeds ($rk^{(0)}_{A\to B}$) between employees as needed. Moreover, the public parameters (*params*) are disclosed within and outside the organization, and are sent to the proxy server. The master key (*msk*) is saved securely in the PKG.

$$
\begin{aligned}
&Hospital\ -PKG:\\
&params = (G_1, G_2, p, e, g, g_1 = g^{\alpha_{HOS}}, g_2, \{w_i\}_0^{2l}, H, H_2)\\
&msk = g_2^{\alpha_{HOS}}\\
&SK_A = (d_{A0}, d_{A1}) = (g_2^{\alpha_{HOS}} H(HOS \parallel A)^{u_A}, g^{u_A})\\
&SK_B = (d_{B0}, d_{B1}) = (g_2^{\alpha_{HOS}} H(HOS \parallel B)^{u_B}, g^{u_B})\\
&rk^{(0)}_{A\to B} = \left(\frac{H(HOS \parallel A)}{H(HOS \parallel B)}\right)^{u_B}
\end{aligned}
$$

$$
\begin{aligned}
&NICT-PKG:\\
&params = (G_1, G_2, p, e, g, g_1 = g^{\alpha_{NICT}}, g_2, \{w_i\}_0^{2l}, H, H_2)\\
&msk = g_2^{\alpha_{NICT}}\\
&SK_A = (d_{C0}, d_{C1}) = (g_2^{\alpha_{NICT}} H(NICT \parallel C)^{u_C}, g^{u_C})
\end{aligned}
$$

Here, $G_1$and G_2within *params* are multiplicative groups with prime order p, where $e: G_1 \times G_1 \to G_2$ is a bilinear map, g is a generator of G_1, g_2 and $\{w_i\}, i = 0,...,2l$ are elements of G_1, w_i is the input of Waters hash function $H: \{0,1\}^{2l} \to G_1$ (for details refer to [4][9]), and H_2 is a hash function $H_2: \{0,1\}^* \to G_2$.

(2) Data-sharing group settings: Groups for sharing data can be created by any user. The creator of the group acts as the group leader. For example, when user A (a doctor at a hospital) creates a group with user B (a doctor of same affiliation) and user C (a health care physician of NICT), in order to execute a health screening project (*job*), A logs in to the proxy server as a leader, and creates a project. Thereafter, the re-encryption key $rk_{A\to B}; ck$ is distributed to the proxy server, and the proxy decryption key (*pdk*) is distributed to C[*1], and the common data ($Y(\tau)$) is sent to the proxy server periodically.

Re-encryption/convert key

$$
\begin{aligned}
(rk_{A\to B}; ck) &= (rk^{(1)}, rk^{(2)}; ck)\\
&= \left(\gamma \cdot H(HOS \parallel A)^{d_\gamma} rk^{(0)}_{A\to B}, g^{d_\gamma}; d_{A1}\right)
\end{aligned}
$$

Proxy decryption key *pdk*

$$
\begin{aligned}
pdk_1^{(job)} &= \gamma \cdot d_{A0} H(job \parallel A)^{d_{job}},\\
pdk_2^{(job)} &= g^{d_{job}}
\end{aligned}
$$

Proxy Common data

$$
y_1(\tau) = \gamma^{-1} H(\tau \parallel A)^{d_\tau}, \quad y_2(\tau) = g^{d_\tau}
$$

Here, γ is a random number taken from G_2, d_γ, d_{job} and d_τ are random numbers taken from Z_p, and are saved securely on the leader's device.

(3) Flexible information sharing: The data sender sets the confidentiality level, encrypts for leader A depending on the confidentiality level, uploads the generated ciphertext $(Enc_K^{AES}(file), Enc_A^{IBPdr}(K))$ to the proxy server, and saves in the A-addressed folder.

$$
\begin{aligned}
Enc_A^{IBPdr}(K) &= (C_0, C_1, C_2, C_3, C_4, C_5)\\
&= (K \oplus H_2(\sigma), \sigma \cdot \hat{e}(g_1, g_2)^r, g^r,\\
&\quad H(HOS \parallel A)^r, H(\tau \parallel A)^r, H(job \parallel A)^r)
\end{aligned}
$$

The confidentiality level is checked by the proxy server. "Medium" and "Low" level data re-encryption is done for user B inside the organization and distributed to B's folder. "Low" level data conversion is also done for user C outside the organization and distributed to C's folder.

proxy server

Re-encryption

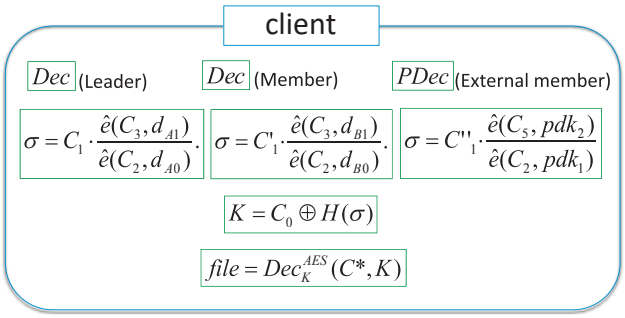
$$
C_1' = C_1 \cdot \left(\frac{\hat{e}(C_4, y_2(\tau))\hat{e}(C_3, rk^{(2)}_{A\to B})}{\hat{e}(C_2, y_1(\tau) \cdot rk^{(1)}_{A\to B})}\right)
$$

Convert

$$
C_1'' = C_1 \cdot \frac{\hat{e}(C_3, ck)\hat{e}(C_4, y_2(\tau))}{\hat{e}(C_2, y_1(\tau))}
$$

Lastly, A, B and C each download data, then A and B do the decryption process (*Dec*), and C does the proxy decryption process (*PDec*), respectively, on their devices.

[*1] When distributing a proxy decryption key, use ID of C to encrypt the *pdk*, then do it via the proxy server.



Remarks: When the leader revokes delegated authority of proxy decryption and re-encryption, he resets the value y and does step (2) once again. In other words, PRINCESS has the flexibility that any user can invoke a project as a leader, add or delete members, and delete and terminate the project. The project deletion and member deletion described here do not only mean deletion of members from the member list, but in addition, they also mean the ability to revoke the re-encryption key and proxy decryption key for the deleted member that were already stored in the server. This is because IBPdr can revoke the approved proxy decryption and re-encryption authority [1]. Therefore, it is possible to terminate not only projects and members, but also cloud storage services when no longer required.

2.3 Performance evaluation

To evaluate the practicality and performance of this system, we prototyped and evaluated implementation. As experimental equipment, for the proxy server we used one Intel Core i7 3770 (3.4 GHz, 64-bit, 32 GB memory), and for the client tablet we used a SONY Xperia Z2 (Android 4.4). As the hash function to be used in the IBPdr library, we implemented the Waters hash function [10], and used the PBC Library (https://crypto.stanford.edu/pbc/) for bilinear map computations. (For details, refer to [1][2]). Measurement results are summarized in Table 1.

Table 1 shows the time required (unit: milliseconds) for a series of processes: "Level check" is the ciphertext confidentiality level check run by the proxy server; "Re-encryption or Convert" is the re-encryption process to the "Medium" and "Low" level data, or the conversion process to the "Low" level ciphertext; "Encryption" is the encryption process carried out when a user uploads information; "Decryption" is the decryption process or the proxy decryption process performed after a user performs download; "Update proxy Common Data" is the creation of common data $Y(\tau)$ carried out by the leader; and "Revocation" is the regeneration of the re-encryption key and proxy decryption key that accompanies user deletion, etc.

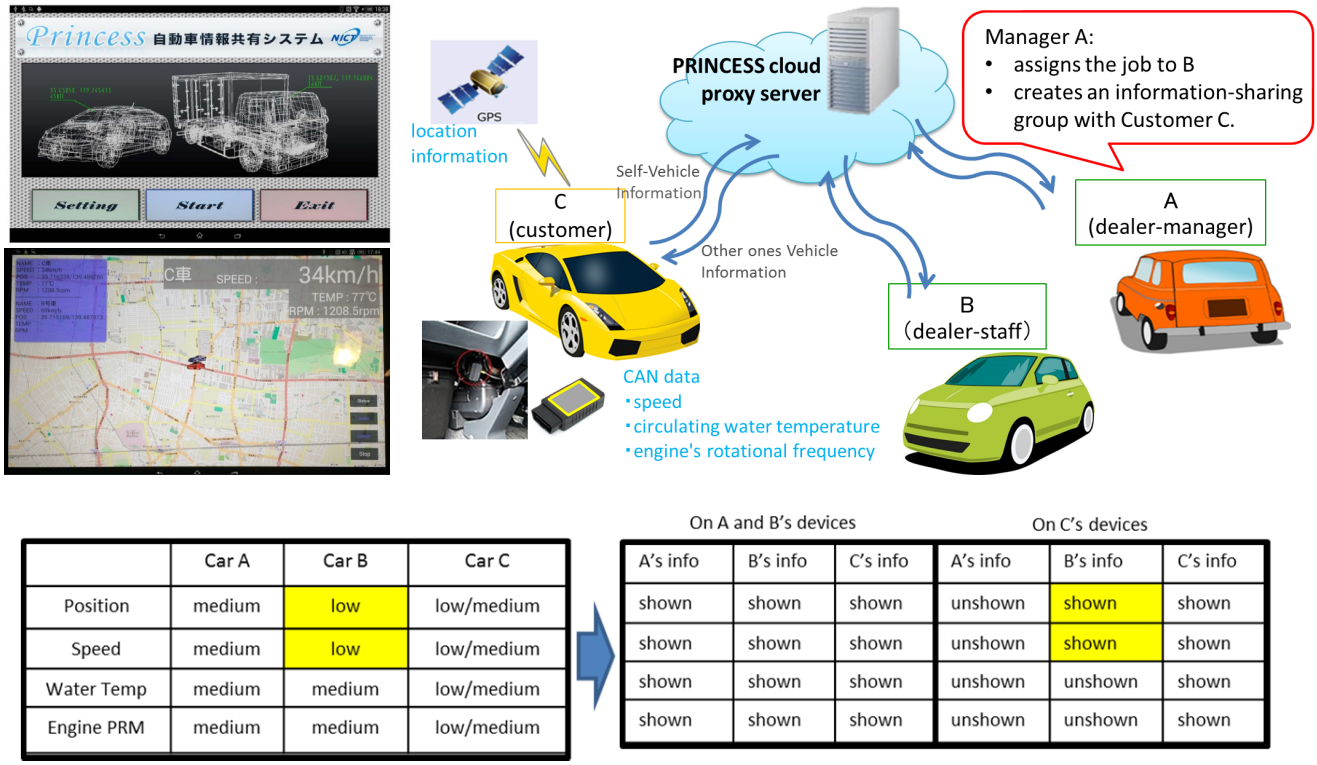
Table 1 Measurement time (milliseconds)

PRINCESS Proxy Server (PC)	
Level check	11.81
Re-encryption or Convert	4.61
Client: any user (Android device)	
Encryption	51.26
Decryption	22.58
Client: group leader (Android device)	
Update proxy Common Data	3.61
Revocation	47.72

3 Application to automobile information sharing system

It is expected that the Internet of Things (IoT) era, i.e., all things are connected to the internet, is coming in the near future. Vehicles are also no exception, and over 1 billion vehicles could be connected to the internet. If the vast data that can be obtained from these vehicles can be collected in the cloud (for example, ITS cloud center), it is forecasted that new big data services can be created. For example, remote access for automobile maintenance and Engine Control Unit (ECU) updates are good examples of this. Furthermore, if GPS information and each type of sensor information are collected on the cloud, then it can be useful for understanding road conditions, not only at normal times but also in times of disasters and emergencies. On the other hand, data obtained from automobiles can contain privacy information, so it requires a mechanism that prevents interception in data transmissions and information leaks from cloud servers. Location information is a typical example of private information, but speed, engine rpm, etc. also reveal driving characteristics of the driver [14]. Furthermore, there is also a movement for including biometric authentication in vehicles, so data obtained from vehicles is expected to also contain much privacy related sensitive information in the future. If this data is collected on the cloud, data confidentiality must be emphasized such that data beyond the required scope does not leak out.

Considering these needs, this section introduces an automobile information sharing system based on PRINCESS (hereinafter referred to as "PRINCESS Automobile Information Sharing System") [5][6].



	Car A	Car B	Car C
Position	medium	low	low/medium
Speed	medium	low	low/medium
Water Temp	medium	medium	low/medium
Engine PRM	medium	medium	low/medium

	On A and B's devices			On C's devices		
	A's info	B's info	C's info	A's info	B's info	C's info
Position	shown	shown	shown	unshown	shown	shown
Speed	shown	shown	shown	unshown	shown	shown
Water Temp	shown	shown	shown	unshown	unshown	shown
Engine PRM	shown	shown	shown	unshown	unshown	shown

Fig. 2 PRINCESS Automobile Information Sharing System (demo)

3.1 System overview

In the PRINCESS Automobile Information Sharing System, location information and CAN data of the vehicle are shared in an encrypted state, and the scope of the sharing destinations of each type of information is set flexibly. With the following scenario as an example, we describe an overview of the system.

Consider a case of the response of a car dealer's manager A and staff member B during a test drive of vehicle C that was repaired by the dealer (Fig. 2). As the dealer's manager, A must obtain location information and CAN data of vehicle C, but A is not required to provide its own vehicle information to customer C. On the other hand, when vehicle C malfunctions and vehicle B is approaching the site, if staff member B provides the location information and speed of vehicle B to customer C, then it seems this would give customer C a sense of safety. When data sharing is summarized based on the settings of confidentiality levels as seen in the table on the lower left of Fig. 2, the result is as shown in the table in the lower right of Fig. 2. To achieve this, the PRINCESS Automobile Information Sharing System shares data as follows. (1) C uploads information encrypted for A, to the proxy server. (2) A distributes to the proxy server the re-encryption key from A to B. With this, B can read information of vehicle C. Then, (3) B uploads to the proxy server the automobile information encrypted for A. (4) A distributes to C a temporarily usable proxy decryption key. With this, C can read the location information of vehicle B.

For a demo, the PRINCESS Automobile Information Sharing System was installed on a SONY Xperia Z2 (Android 4.4) tablet, and a test done where the PRINCESS Automobile Information Sharing System shares the location information obtained from GPS in real time from the vehicle when actually driving, as well as CAN data obtained via the OBD II port such as vehicle speed, cooling water temperature, and engine RPMs.

3.2 Application fields of the PRINCESS automobile information sharing system

The scenario described above is between the dealer and customer, but the PRINCESS automobile information sharing system can also be applied to various scenarios.

Drive with friends: By sharing location information and traffic jam information while driving with friends, they can adjust the meeting time and place flexibly.

Operations management of transportation companies: It is possible to collect and share location information and CAN data within the company, then use this for adminis-

trative tasks such as car dispatch and car maintenance.

Auto theft prevention: It is possible to check for thefts by sending encrypted location information at a "Low" confidentiality level from the car to one's mobile phone, etc. Moreover, by sharing this information with the police, it is possible to help find stolen cars.

Safe driving attestation for automobile insurance: Vehicle information containing CAN data, etc. is encrypted at a "Low" confidentiality level, which is uploaded to a proxy server, then provided to the insurance company when needed. By a member revocation function, flexible sharing is possible, with information sharing for only the fixed time of an estimation, etc.

4 Future prospects and issues

PRINCESS is a general purpose encryption file sharing storage system that enables various applications, such as shared systems of medical data and automobile information, utilization of social big data, etc. This information probably includes private information, so there are security concerns about using cloud storage services that lack insufficient data protection measures. PRINCESS not only solves such privacy problems, but can also be a useful system from the point of view of Business Continuity Planning (BCP). We will continue working toward practical implementation of PRINCESS and the PRINCESS Automobile Information Sharing System that are used in various fields.

Acknowledgments

We are grateful to Technical Researchers Takashi Kurokawa and Sachiko Kanamori of the NICT Security Fundamentals Laboratory for their huge contributions by examining system specifications, support in paperwork procedures, etc.

References

1 L. Wang, A. Waseda, R. Nojima, S. Moriai: "PRINCESS: Proxy Re-encryption with INd-Cca security in Encrypted file Storage System", SCIS2014, Jan. 2014.
2 L. Wang, T. Hayashi, S. Kanamori, A. Waseda, R. Nojima, and S. Moriai: "POSTER: PRINCESS: A Secure Cloud File Storage System for Managing Data with Hierarchical Levels of Sensitivity", CCS2015, pp.1684-1686, ACM 2015.
3 L. Wang: "The Method and its Cryptosystem of the ID-based Encryption with Two Functions (Proxy Decryption and Proxy Re-encryption)", Patent no. 5298394.
4 L. Wang, L. Wang, M. Mambo, and E. Okamoto: "Identity-Based Proxy Cryptosystems with Revocability and Hierarchical Confidentialities", IEICE, E95-A(1), pp.70-88, 2012.
5 L. Wang, R. Nojima, S. Moriai: "PRINCESS-based Secure Automobile Information Sharing System", SCIS2015, Jan. 2015.
6 L. Wang, R. Nojima, S. Moriai: "A Secure Automobile Information Sharing System", AsiaCCS-IoTPTS2015, pp.19-26, ACM 2015.
7 A. Shamir: "Identity-based cryptosystems and signature schemes", In: CRYPTO 1984, LNCS 196, pp.47-53, Springer 1985.
8 D. Boneh, X. Boyen: "Efficient Selective Identity-Based Encryption Without Random Oracles", J.Cryptology 24(4), pp.659-693, 2011.
9 B. Waters: "Efficient Identity-Based Encryption Without Random Oracles," EUROCRYPT 2005, pp.114-127, 2005.
10 M. Mambo, and E. Okamoto: "Proxy cryptosystem: Delegation of the power to decrypt ciphertexts", IEICE, E80-A(1), pp.54-63, 1997.
11 M. Blaze, G. Bleumer, and M. Strauss: "Divertible protocols and atomic proxy cryptography", In: EUROCRYPT 1998, LNCS 1403, pp.127-144, Springer 1998.
12 B. Libert, D. Vergnaud: "Unidirectional Chosen-Ciphertext Secure Proxy Re-encryption", In: PKC 2008. LNCS 4939, pp.360-379, Springer, 2008.
13 R. Hayashi, T. Matsushita, T. Yoshida, Y. Fujii, and K. Okada: "Unforgeability of Re-Encryption Keys against Collusion Attack in Proxy Re-Encryption", In: IWSEC 2011, LNCS 7038, pp.210-229, 2011.
14 A. Waseda, R. Nojima: "On the experiment of privacy leakage analysis in a vehicle to vehicle communication", CSS2015, Jan. 2015.

Lihua WANG, Ph.D.
Senior Researcher, Security Fundamentals Laboratory, Cybersecurity Research Institute
Design and Analysis of Cryptographic Protocols

Takuya HAYASHI, Ph.D.
Researcher, Security Fundamentals Laboratory, Cybersecurity Research Institute
Cryptanalysis, Efficient Implementation

Atsushi WASEDA, Ph.D.
Senior Researcher, Security Fundamentals Laboratory, Cybersecurity Research Institute
Information Security

Ryo NOJIMA, Ph.D.
Senior Researcher, Security Fundamentals Laboratory, Cybersecurity Research Institute
Cryptography, Cryptographic Protocol

Shiho MORIAI, Ph.D.

Director of Security Fundamentals
Laboratory, Cybersecurity Research Institute
Cryptography, Information Security