Proceedings of APCC2008 copyright © 2008 IEICE 08 SB 0083

A Secure Lightweight Upload Protocol for Single-use RFID

Koki MITANI, Koichi TAKASUGI, Mika ISHIZUKA[1], Hideki TOSHINAGA,
Hiroshi SHIBATA, Satoshi KOTABE, Hiroshi TOHJO[1], and Hiroshi SAITO
NTT Network Innovation Laboratories
mitani.koki@lab.ntt.co.jp

***Abstract*-We propose an upload protocol for single-use RFID for achieving wireless single-use applications such as notification of an opened envelope. The single-use RFID proposed here has the following features: an RFID to work actively, a limited number of data transfers, low-bit data transfer, and long-range radio-communication by VHF/UHF frequency. For single-use RFID applications, the upload protocol should be implementable on a small circuit and secure from remote attacks. Therefore, we propose pre-computation of transmitting/receiving frames and integration of message exchanges to reduce computation and communication without losing security. An evaluation showed that the amount of computation and communication of RFID was reduced by the proposed protocol.**

Keywords: RFID, Security, Single-use, Ubiquitous Network, Wide-Area Ubiquitous Network, WAUN

I. INTRODUCTION

Over the last several years, ubiquitous computing has been integrated into many aspects of our lives because of the evolution of radio frequency identification (RFID) and mobile phone technology. However, there are still many non-networked objects in the real world. Envelopes, umbrellas, and bicycles are examples of non-networked objects; they are too small or too distributed for attaching a networked processing device such as an RFID or a mobile phone. There is a large number of non-networked objects (for example, there were more than 10 billion envelopes transported by the Japan Post Service in fiscal 2006 [1]). By networking a large number of objects having small amounts of distributed information, we will achieve a truly ubiquitous computing life.

For example, to notify a sender of the opening of an envelope, an RFID attached to an envelope needs to transfer 0–1-bit data once. Therefore, we focus on the limited number of communications and limited amount of data transfer of single-use applications, such as notification of the opening of an envelope, and propose an upload protocol for a single-use RFID.

For this study, we define a single-use RFID as a networked processing device with the following features: an RFID to work actively, a limited number of data transfers, low-bit data transfer, and long-range radio-communication by VHF/UHF frequency. The network topology between RFIDs and databases is shown in Fig. 1. That network consists of more RFIDs than people, each RFID reader within a few kilometers radius, and of 10–100 databases. The link between an RFID and RFID reader is wireless, while the link between an RFID reader and a database is wired. A Wide Area Ubiquitous Network (WAUN), which is the concept of a "long wireless link with low-power-consumption low-end terminals" [2], is one of the applicable network systems. The wireless link of WAUN has a large range (about 5 km), which enables a network provider to cover a wide area with a small number of RFID readers and thus offer a service at a reasonable cost.

The upload protocol for a single-use RFID requires the following features: high security, lightweight computation, and narrow-band communication. Because the distance between a single-use RFID and RFID reader is a few kilometers, four kinds of security (anonymity, authentication, confidentiality, and integrity) are required to protect against remote attacks by tracking, spoofing, eavesdropping, and altering. Furthermore, a small circuit and small battery are required to implement a single-use RFID on a small mobile object such as an envelope.

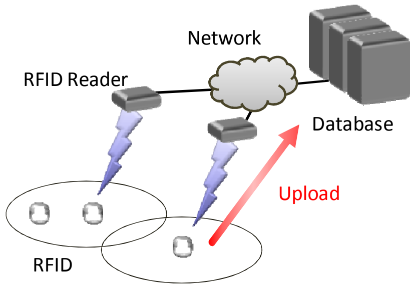

Figure 1. Network topology between RFID and database

II. RELATED WORK

For achieving high security RFID with existing technologies, several lightweight security protocols must be integrated. In this section, we explain existing security protocols for RFID and integrate them into one conventional protocol, the message sequence of which is shown in Fig. 2.

A. Anonymity

Anonymity is for preventing tracking. According to a survey paper [3], there is a tree approach, a synchronization approach, and a time-space tradeoff approach for RFID that is capable of computing symmetric-key functions. In the case of an active tag such as a single-use RFID, a synchronization approach [4] is appropriate for the conventional protocol because of its design for simple RFID implementation. Generating an anonymous ID at both sides of the wireless link is required. An anonymous ID is generated by hashing an ID hash seed, and an ID hash seed is generated by hashing the previous ID hash seed repeatedly. The number of

[1] In 2008, Mika ISHIZUKA moved to NTT Advanced Technology Corporation and Hiroshi TOHJO moved to NTT BizLink, Inc.

times to hash an ID hash seed is stored as an ID hash parameter at both sides.

B. *Authentication*

Authentication is for preventing spoofing. There is much research on authenticating RFIDs and RFID readers by a challenge-response-based protocol. In the case of a single-use RFID, an RFID reader-initiated 3-way wireless message sequence [5] is appropriate for efficient use of wireless bandwidth. Generating a challenge text and generating and validating an authentication token at both sides of the wireless link are required. An authentication token is generated by hashing the challenge text and secret information.

C. *Confidentiality*

Confidentiality is for preventing eavesdropping. There is a low-cost implementation of standard encryption/ decryption algorithms, such as AES, for RFID [6]. Storing secret information, sequence numbers, and cryptographic processing (encryption and decryption) of transferring data is required.

D. *Integrity*

Integrity is for preventing alterations. In general, generating and validating a message authentication code is required [7]. A message authentication code is generated by hashing a sequence number, the transferring data, and the secret information.

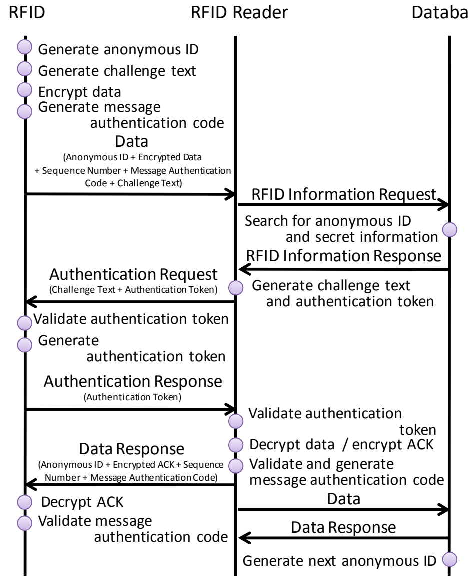
RFID
RFID Reader
Database
Generate anonymous ID
Generate challenge text
Encrypt data
Generate message authentication code
Data
(Anonymous ID + Encrypted Data + Sequence Number + Message Authentication Code + Challenge Text)
RFID Information Request
Search for anonymous ID and secret information
RFID Information Response
Authentication Request
(Challenge Text + Authentication Token)
Generate challenge text and authentication token
Validate authentication token
Generate authentication token
Authentication Response
(Authentication Token)
Validate authentication token
Data Response
(Anonymous ID + Encrypted ACK + Sequence Number + Message Authentication Code)
Decrypt data / encrypt ACK
Validate and generate message authentication code
Data
Decrypt ACK
Validate message authentication code
Data Response
Generate next anonymous ID

Figure 2. Message sequence of conventional protocol

As a result, the conventional protocol has high security features (anonymity, authentication, confidentiality, integrity), and its computation algorithm and message exchange are well designed for existing RFIDs that communicate continuously. However, it is not suitable for a single-use application, which needs only a few small data transfers, because separate message exchanges lead to increased message exchanges, and on-demand security computation leads to increased computation.

III. Upload Protocol for Single-use RFID

We propose a new upload protocol especially designed for single-use RFID. Our protocol optimizes the conventional protocol for single-use RFID by integrating authentication and data transfer messages and pre-computing transmitting/receiving frames, without losing security features.

The message sequence of the proposed protocol is shown in Fig. 3. As described later, an RFID and a database share pre-computed frames. When the RFID uploads data, it simply searches for a pre-computed transmitting frame associated with a sequence number and data string and transmits that frame (Authentication+Data Message). An RFID reader receives that whole wireless frame and then sends an inquiry with that frame to the database for retrieving RFID information, including ID and data string (RFID Information Request Message). The database searches for the RFID information by the whole wireless frame and then sends that information to the reader (RFID Information Response Message). If the reader receives valid RFID information from the database, it transmits an acknowledgement to the RFID (Authentication+Data Response Message) and sends the data to the database (Data Message). When the RFID receives acknowledgement from the network, it compares the received frame with the stored expected frame to validate that frame. If the received frame is invalid, the RFID transmits the same data frame again. When the database receives the data, it stores it and sends an acknowledgement to the RFID reader (Data Response Mesage).

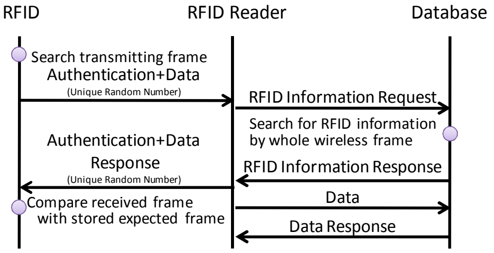
RFID
RFID Reader
Database
Search transmitting frame
Authentication+Data
(Unique Random Number)
RFID Information Request
Search for RFID information by whole wireless frame
Authentication+Data Response
(Unique Random Number)
RFID Information Response
Compare received frame with stored expected frame
Data
Data Response

Figure 3. Message sequence of upload protocol for single-use RFID

A. *Message Integration*

Message integration is one of two optimizations. In the proposed sequence, the authentication message exchange and data message exchange are integrated into one message exchange.

The conventional protocol (Fig. 2) has separate message exchanges for authentication and data transfer because it assumes that the RFID transfers more than two data frames in a short time. If the RFID transfers many data frames in a short time, this separation enables authentication to be efficiently merged into the first message exchange and the total wireless message exchanges is reduced. However, if a single-use RFID transfers only one frame in a short time, this separation leads to increased message exchanges. Therefore, we propose integrating the authentication and data transfer messages (Fig. 3).

B. *Pre-computation*

Pre-computation is the other optimization for the proposed protocol and can eliminate unnecessary computation for continuously generating parameters, without losing security features.

The conventional protocol (Fig. 2) must compute security parameters to construct transmitting frames and to validate received frames. Computation for generating anonymous ID and authentication tokens supports a number of continuous secure authentications. Computation for encrypting and generating message authentication codes supports a number of continuous secure data transfers. However, if a single-use RFID transfers only a few low-bit data frames, we can pre-compute all the expected transmitting/receiving frame patterns and store them in its permanent memory. Therefore, in the proposed protocol (Fig. 3), security processing is replaced by the search for and comparison of frames.

The pre-computed frame is shown in Fig. 4. We use a secret bit string of a unique random number as a pre-computed frame. To equip the wireless frame with the same data transfer functions and security as the conventional protocol, the secret bit string should be the same length as the conventional wireless frame, including ID, authentication token, data, sequence number, and message authentication code. If the implementation has the following features, it is as secure as a one-time pad: a secure random number generation algorithm, secure offline exchange of a random number between the RFID and RFID reader/database, and prevention of the reuse of the random number [7]. In addition, in the case that an adversary uses an arbitrary random bit string to attack, the low probability of success by pure coincidence is the same as in the conventional protocol because the length of the bit string is the same.

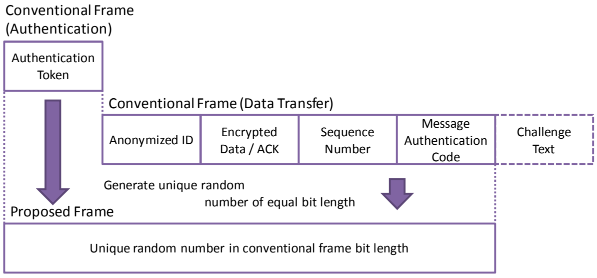

Figure 4. Pre-computed wireless message frame of upload protocol

An example of pre-computed frame allocation is shown in Table I. If the RFID is designed for 2 1-bit data transfers, it must store 8 pre-computed unique random numbers in its permanent memory (4 possible transmitting frames and 4 expected receiving frames). In this case, the actual number of frames to be used is 4 because the actual transmitting data is only one of two digits (0 or 1).

TABLE I
EXAMPLE OF PRE-COMPUTED FRAME ALLOCATION

Sequence Number	Data	Possible Transmitting Frames	Expected Receiving Frames
1	0	Random Number 1	Random Number 2
1	1	Random Number 3	Random Number 4
2	0	Random Number 5	Random Number 6
2	1	Random Number 7	Random Number 8

IV. EVALUATION

We compared the proposed protocol with the conventional protocol by simulating the number of the wireless frame, total amount of information in the wireless frame, amount of stored information, number of levels of nodes in a binary search tree, and security computation by the RFID and RFID reader/database. The measurement targets are shown in Fig. 5.

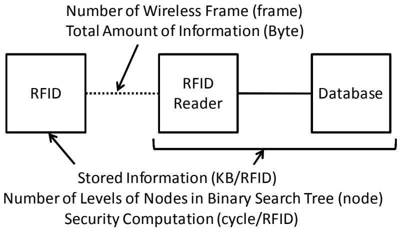
Number of Wireless Frame (frame)
Total Amount of Information (Byte)
RFID
RFID Reader
Database
Stored Information (KB/RFID)
Number of Levels of Nodes in Binary Search Tree (node)
Security Computation (cycle/RFID)

Figure 5. Measurement Targets

The envisioned parameters of single-use RFID are shown in Table II. A single-use RFID can transfer 3-bit data thrice. The acceptable number of gates in RFID is 5 000–10 000, the acceptable memory size is 0.2–100 KB, and the number of gates available for security computation is estimated to be below 5 000 [8]. The envisioned number of RFID devices is 10 billion, which is greater than the world population.

TABLE II
ENVISIONED PARAMETERS OF SINGLE-USE RFID

Parameters	Value
Data Length	0–3-bit
Number of sequences	1–3
Number of Gates	5 000–10 000
Memory	0.2–100 KB
Number of RFID	10 billion

The security parameters and algorithms for this evaluation are shown in Table III. We adopted SHA-256 (hash), HMAC-SHA-256 (message authentication), and AES-128 (encryption and decryption) because those security algorithms satisfy the 128-bit security that the National Institute of Standards and Technology (NIST) recommends for US information systems after 2030 [9]. In addition, for simulating the computation amount of each security algorithm, we used the evaluation parameters determined by the New European Schemes for Signature, Integrity, and Encryption (NESSIE) [10] on a Pentium 4 1.7-1.8GHz/Linux computer.

TABLE III
PARAMETERS AND ALGORITHMS FOR SIMULATION

	Parameters and Algorithms	Value
Bit-Length	ID (Same ID space as IPv6 protocol)	16 byte
	Challenge Text for Authentication	8 byte
	Authentication Token (SHA-256)	32 byte
	Message Authentication Code (HMAC-SHA-256)	32 byte
	Sequence Number (Max number of sequences: 256)	1 byte
Parameter	ID Hash Seed (SHA-256)	32 byte
	ID Hash Parameter (Max number of sequences: 256)	1 byte
	Secret for Authentication (SHA-256)	32 byte
	Secret for Message Authentication (HMAC-SHA-256)	64 byte
	Secret for Encryption (AES, 128-bit)	16 byte
Computation	Hash (SHA-256)	40 cycles/byte
	Message Authentication Code Generation (HMAC-SHA-256)	40 cycles/byte
	Encryption (AES, 128-bit)	24 cycles/byte
	Decryption (AES, 128-bit)	25 cycles/byte

A. Wireless Security and Resource Consumption

A comparison of the total amount of information and number of frames for a whole data transfer session is shown in Fig. 6. Both protocols need about a 200-byte information transfer, but the proposed protocol transfers only half the number of frames and does not need to exchange challenge texts.

If the wireless bandwidth is 30% of 5 channels of 9 600 bit/s, more than 100 million RFIDs under one RFID reader can transfer data thrice a year. In addition, the proposed protocol can use the bit string more efficiently because the boundary between authentication- and data-related fields (Fig. 4) can be eliminated.

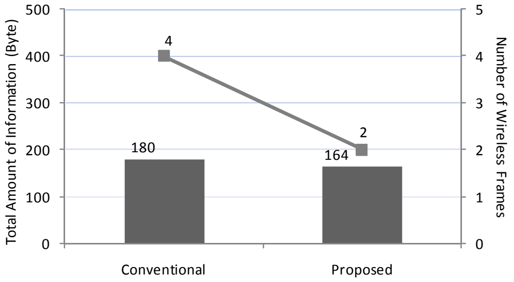

Figure 6. Total amount of information and number of wireless frames

B. *Reduction of Implementation Complexity*

In this section, we evaluate the reduction of implementation complexity in RFID.

The amount of security computation by an RFID in one sequence is shown in Fig. 7. The conventional protocol needs security computation for continuous communication, but the proposed protocol does not need security computation for each sequence because it has been pre-computed. Hardware implementation of cryptographic processing such as AES takes on 3 000–30 000 gates, and a cryptographic hash function such as SHA-1 takes on 2 000–20 000 gates [8].

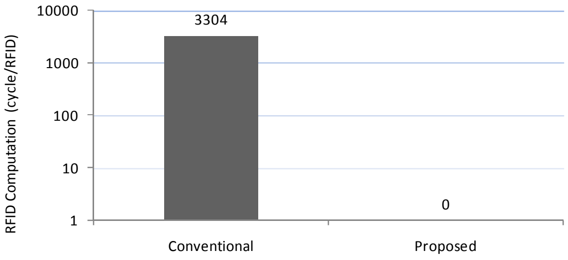

Figure 7. RFID computation (n-bit of data: 1–8-bit)

The number of levels of nodes beneath the root in a self-balancing binary search tree of stored frames in RFID is shown in Fig. 8. Unlike the conventional protocol, the proposed protocol needs to search this tree and compare frames for each data transfer. When the data variation or number of sequences increases, the number of searches and comparisons increases linearly. However, such processing can be implemented as a combination of primitive bitwise operations (e.g. AND and XOR).

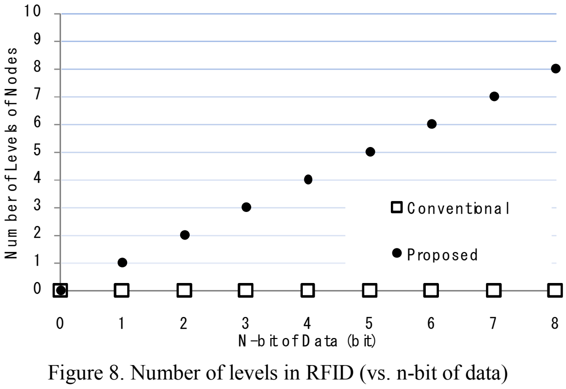

Figure 8. Number of levels in RFID (vs. n-bit of data)

Therefore, the proposed protocol can replace 5 000–50 000 gates by combining primitive bitwise operations that can be implemented with a small number of gates. Furthermore, its computation for generating and validating frames is lower than in the conventional protocol because the proposed protocol needs only half the number of frames as the conventional protocol does (Fig. 6).

The number of levels of nodes beneath the root in the self-balancing binary search tree of stored frames in the RFID reader/database is shown in Fig. 9. Because the increase of searches depends on the O(logN) of the product of the data variation and number of sequences, the increase of searches in the proposed protocol is less than 15% .

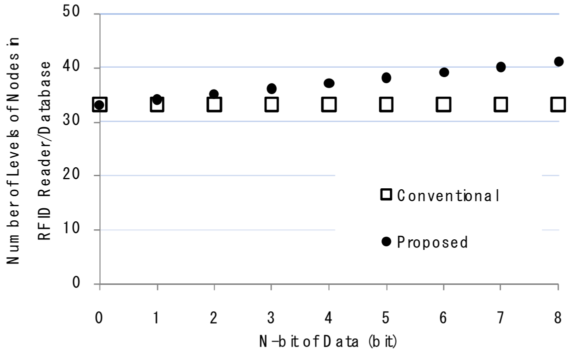

Figure 9. Number of levels in RFID reader/database (vs. n-bit of data)

As a result, we could reduce the complex security computation of RFID to less than 5 000 gates (bitwise operations). Reducing the gates reduces the circuit size, and reducing the computation reduces the battery size. Reducing the security computation by RFID readers/databases decreases the number of RFID readers and databases needed.

C. *Impact of Increasing Stored Information*

In this section, we evaluate the impact of increasing the stored information, which is the main drawback of the proposed protocol. Table IV shows the stored information of the RFID and RFID reader/database.

TABLE IV
STORED INFORMATION

	Conventional Protocol	Proposed Protocol
RFID	ID Hash Seed ID Hash Parameter Secret for Authentication Secret for Message Authentication Secret for Encryption Sequence Number	Possible Transmitting Frames Expected Receiving Frames
RFID Reader and Database	ID ID Hash Seed ID Hash Parameter Secret for Authentication Secret for Message Authentication Secret for Encryption Sequence Number	Possible Transmitting Frames Expected Receiving Frames

A comparison of the amount of stored information in RFID with the different protocols is shown in Figs. 10 and 11. Instead of storing various security parameters as an RFID in the conventional protocol does, an RFID in the proposed protocol stores only transmitting/receiving frames (Table IV). In the proposed protocol, the number of possible transmitting frames and expected receiving frames depends on data variation and the number of sequences. The horizontal axis of Fig. 10 is n-bit of data. When n-bit of data increases, the stored information increases exponentially (2^n). The horizontal axis of Fig. 11 is the number of sequences. When the number of sequences increases, the stored information increases linearly. In general, an RFID has 0.2–100 KB of memory. A small RFID with 0.2 KB of memory can transfer zero-bit data once, and a large RFID with 50 KB of memory can transfer 8-bit data or communicate more than 300 times with the proposed protocol.

A comparison of the amount of stored information in RFID readers/databases with the different protocols is shown in Fig. 12. As in the results for an RFID, the number of possible transmitting frames and expected receiving frames depends on data variation and the number of sequences. Transferring more than 3-bit data thrice with the proposed protocol takes 10 times as much disk space as with the conventional protocol. However, the information for 10 billion RFIDs could be stored in only 100 databases with 400 GB of disk space each.

Although the stored information and number of searches rose, they have only a limited effect for RFID and RFID readers/databases that transfer only a few low-bit data frames.

V. Conclusion

We can reduce security computation without losing security features by message integration and pre-computation. This enables smaller RFID circuits and batteries. Although stored information and the number of searches increase, they have only a limited effect for a single-use RFID and its reader/database. Therefore, the proposed protocol could be applied for developing single-use applications using single-use RFID.

References

[1] Ministry of Internal Affairs and Communications, Japan, “Information and Communications in Japan: White Paper 2007,” Jul. 2007

[2] H Saito and K Takasugi, “Recent developments in wide area ubiquitous network research,” *The 8th International Symposium on Autonomous Decentralized Systems (ISADS 2007)*, Mar. 2007, pp. 503–507.

[3] A. Juels, “RFID security and privacy: a research survey,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, Feb. 2006, pp. 381–394.

[4] M. Ohkubo, K. Suzuki, and S. Kinoshita, “Cryptographic approach to “privacy-friendly” tags,” *RFID Privacy Workshop*, Nov. 2003.

[5] M. Aigner and M. Feldhofer, “Secure symmetric authentication for RFID tags,” *Telecommunications and Mobile Computing TCMC2005*, Mar. 2005.

[6] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, “Strong authentication for RFID systems using the AES algorithm,” *Workshop on Cryptographic Hardware and Embedded Systems CHES 2004*, vol. 3156, Aug. 2004, pp. 357–370.

[7] B. Schneier, “Applied cryptography: protocols, algorithms, and source code in C, second edition,” Wiley, 1996.

[8] M. Lehtonen, T. Staake, F. Michahelles, and E. Fleisch, “From identification to authentication—a review of RFID product authentication techniques,” *Workshop on RFID Security (RFIDSec)*, Jul. 2006.

[9] NIST, “Recommendation on Key Management NIST Special Publications 800 Series, SP 800-57,” 2005.

[10] B. Preneel et al., “Final report of NESSIE, New European Schemes for Signatures, Integrity, and Encryption,” LNCS Springer-Verlag, in press, pp. 350–473.

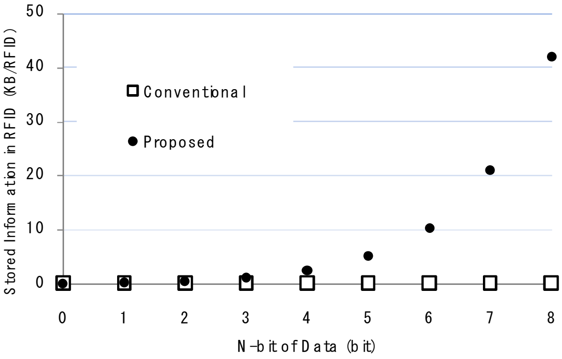

Figure 10. Stored information in RFID (vs. n-bit of data)

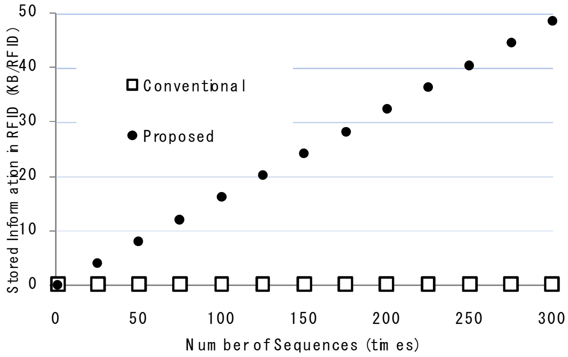

Figure 11. Stored information in RFID (vs. number of sequences)

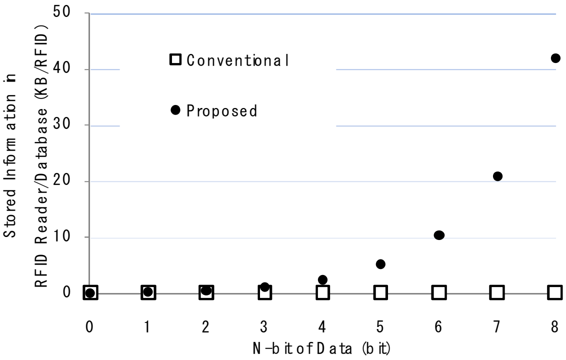

Figure 12. Stored information in RFID reader/database (vs. n-bit of data)