LETTER

Privacy-Preserving RFID Authentication Using Public Exponent Three RSA Algorithm

Yoonjeong KIM[†a)], *Member*, SeongYong OHM[††], *and* Kang YI[†††], *Nonmembers*

SUMMARY In this letter, we propose a privacy-preserving authentication protocol with RSA cryptosystem in an RFID environment. For both overcoming the resource restriction and strengthening security, our protocol uses only modular exponentiation with exponent three at RFID tag side, with the padded random message whose length is greater than one-sixth of the whole message length.

key words: *cryptography, security, security of data*

1. Introduction

Much research has been done concerning lightweight cryptographic functionality on Radio Frequency IDentification (RFID) tags. Privacy-preserving authentication, where only authenticated users can see the tag IDs, is one of the essential issues that needs to be resolved, and it is also considered a must for the widespread use of RFID. A standard asymmetric algorithm based on cryptography, such as RSA, can be used for the authentication. However, implementing an RSA cryptosystem in RFID tags is regarded as entirely infeasible due to its resource overhead [1].

In this letter, however, we propose a privacy-preserving authentication protocol for RFID tags using an RSA cryptosystem, which can be considered an effort to overcome the recognized infeasibility of this approach. First, we use the value of public exponent as three, thus have $C = M^3 \ mod \ N$ as encryption algorithm where M is plain text, C is cipher text and N is modulus. We strengthen the security of our system by padding a random message whose length is greater than one-sixth of the whole message length in order to make the RSA algorithm secure for the low exponent.

2. RFID Privacy Preserving Authentication Protocol

The proposed protocol is based on the RSA asymmetric key authentication communication between the Tag and the Reader [2]. The private key for the user who can see the RFID tag ID is stored on the Reader side. RFID tag ID ID_{tag}, its related key K_{tag}, and public key of the user are stored in an RFID tag side before actual authentication. The authentication is processed as follows, as shown in Fig. 1.

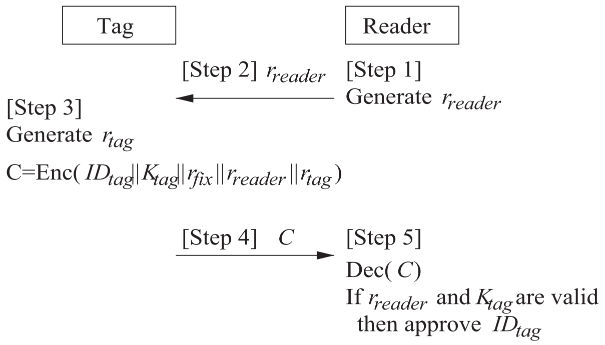

Fig. 1 Privay-preserving authentication protocol for RFID tag based on public-key cryptosystem.

Step 1 The reader generates a random r_{reader}.
Step 2 The reader requests identification to the tag with r_{reader}.
Step 3 The tag generates r_{tag} and calculates $C = Enc(ID_{tag} \| K_{tag} \| r_{fix} \| r_{reader} \| r_{tag})$ using the stored public key of the reader, where r_{fix} is a fixed number initially set as a random number, and || denotes concatenation.
Step 4 The tag sends the encrypted message C to the reader.
Step 5 The reader decrypts the received message C using its private key, checking if r_{reader} is equal to the value sent in step 1, and if K_{tag} is equal to the key of ID_{tag} stored in the database on the Reader side.

In this protocol, only the authenticated users can get the RFID tag ID. Any other users without the private key cannot decrypt the encrypted message, so they cannot get the RFID tag ID. Consequently, *the protocol satisfies the privacy-preserving authentication.*

3. Why We Use Public Key as $e = 3$

RFID tags have limited area, time and power consumption. Area insufficiency can be solved by a scalable design [3]–[5]. The time limit can also be solved by using faster clocks. The power limit is the actual problem that we should solve.

The current passive RFID tag chip has a maximum of 10 μW for power budget for both analog and digital parts. From among the 10 μW, the digital part can get 1 μW for its processing [6]. The power consumption for a single 1024-bit modular multiplication operation is about 0.48 μW as-

Manuscript received July 3, 2008.
Manuscript revised October 28, 2008.

[†] The author is with the Department of Information Security, Seoul Womens University, Seoul, 139–774, Korea.

[††] The author is with the Computer Science and Engineering, Seoul Womens University, Seoul, 139–774, Korea.

[†††] The author is with the School of Computer Science and Electrical Engineering, Handong Global University, Pohang, 791–708, Korea.

a) E-mail: yjkim@swu.ac.kr

DOI: 10.1587/transinf.E92.D.545

suming 1.2 volt supply voltage [4]. Therefore the upper boundary limit for the number of modular multiplication is two. This means that, considering the power budget of RFID's Tag side, exponent 3 is the only feasible choice for us.

As the public exponent value becomes larger, the security gets strengthened. However, as we show in the above, the public exponent $e = 3$ is the only choice for considering real RFID tag chip design and implementation [4]–[8].

The implementation of pseudo-random number generator on RFID tag is also possible as Appendix A.

4. Security Analysis of the Protocol

4.1 Security against the Known Attacks

Now, we inspect the security of the protocol against known attacks. First, eavesdropping attack cannot succeed because the exchanged messages are random patterns or encrypted data. Second, reader spoofing cannot be done because only those who know the private key can decrypt the message. Third, tag spoofing is also ineffective because no one except the real tag knows K_{tag}, so a spoofed tag cannot generate the correct encrypted message. Finally, replaying of message r_{reader} at Step 2 has no effect because RFID tag generates another random number r_{tag} for encrypting the message each time. Replaying of message C at Step 4 has no effect because C includes r_{reader} which is generated differently whenever an authentication is requested.

4.2 Security against Partial Key Exposure Attacks

The partial key exposure attacks are based on the fact that attackers can gain access to partial information of the private exponent d such as the most significant bits or least significant bits [9]–[11]. As shown in Fig. 1, in the proposed protocol, the private key is used only on the Reader side. Generally, the Reader side is considered more secure than the Tag side, so, the proposed protocol is more secure than other authentication protocols.

4.3 Security against Power Attacks

Power analysis analyze power consumption measurements to find secret keys from tamper resistant devices [12]. For modular exponentiation operations, it is possible to test exponent bit guesses by testing whether predicted intermediate values are correlated to the actual computation. We can note that the low exponent that is used in the proposed protocol is the public key, not the secret key. Thus, according to our research results, it is possible to say that power attacks are not a concern in the proposed protocol.

4.4 Security of RSA Cryptosystem with Public Key e=3

Significant weaknesses of the RSA algorithm with a low public exponent is owing to Hastad's broadcast attack and Franklin-Reiter's related message attack [13]. Hastad's attack happens when a sender sends an encrypted message M to a number of parties $P_1, P_2, \ldots, P_k$ and each party has its own RSA key (N_i, e_i). Fraklin-Reiter's attack occurs when a sender sends to a receiver related encrypted messages using the same modulus. Hastad's attack is not applicable to our proposed protocol since the messages in our protocol are processed with a party which has one RSA key. Franklin-Reiter's attack cannot be effective, either, because the messages are unrelated – they include random numbers. Another meaningful attack on the RSA algorithm with a low public exponent is Coppersmith's which uses a solution smaller than $N^{1/K}$ to polynomial equation ($modN$) of degree k in a single variable x. Coppersmith also suggested the strengthening method against the attack, which will tolerate the attack if the length of random padding is greater than or equal to one-sixth of the whole message [13], [14]. For 1024-bit modulus N, the secure random padding length is just 171 bits or more. The input format of $Enc()$ at Step 3 is shown in Fig. 2. The displayed bit lengths in Fig. 2 show the recommended field lengths guaranteeing the security of the RSA with 1024 bit operands.

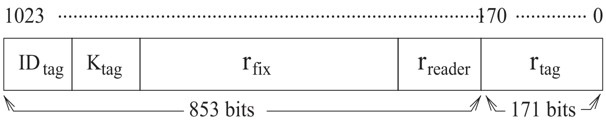

Fig. 2 The input format of Enc() at Step 3: For 1024-bit operands, the recommended bit length of r_{tag}, considering security, is 171 bits.

The random padding might become more secure through the spreading of random padding [14]. This is spreading the random padding into several blocks (not one continuous block). For example, two bits out of each eight-byte - this seems to be a much more effective defense against the attack.

5. Pros. and Cons. of the Proposed Method Compared to Possible Variants

The possible variants of the proposed method might be using Sun and Wu's Rebalanced RSA-CRT (Chinese Remainder Theorem) [16], or ECC (Elliptic Curve Cryptography) [17].

Rebalanced RSA-CRT further speeds up RSA-CRT decryption by shifting decryption cost to encryption cost. Sun and Wu have designed a variant of Rebalanced RSA-CRT, and the variant has the public exponent $e = 2^{511} + 1$ such that its encryption is faster than the original RSA-CRT. The environment of our proposed method has limited resources for public key encryption. Thus, the faster work of decryption is not a concern in the proposed protocol.

ECC is a crytosystem that was proposed by Koblitz and Miller in 1985, and operates on groups of points over elliptic curves and derives its security from the hardness of the elliptic curve discrete logarithm problem. Although ECC could still be a useful tool for some applications, it is unlikely it will supplant RSA. It is held that customers should choose

RSA when feasible, but consider ECC when private key operations emerge as a performance bottleneck [17]. The environment of our proposed method has limited resources for only public key encryption, not private key encryption - private key encryption is done on the Reader side which is capable of a more powerful performance.

In summary, the proposed method is more adaptable in ubiquitous computing environments where public key encryption is done on limited resources, while the method is not adaptable in a general environment, compared to RSA-CRT. Also, the proposed method is more widely used though not useful when the private key encryption is done at limited resources, compared to ECC.

6. Conclusions and Discussions

In this letter, we propose a feasible solution of RFID privacy-preserving authentication protocol based on the RSA cryptosystem. This work can be applied for the high-privacy preserving RFID applications such as electronic passports. Our results reflect the first trial of successfully using an RSA cryptosystem in RFID applications, which has previously been regarded as infeasible.

Acknowledgement

This research is supported by Foundation of ubiquitous computing and networking (UCN) Project, the Ministry of Knowledge Economy (MKE) 21st Century Frontier R&D Program in Korea.

References

[1] A. Juels and S.A. Weis, "Authenticating pervasive devices with human protocols," CRYPTO, V. Shoup, ed., Lecture Notes in Computer Science, vol.3621, pp.293–308, Springer, 2005.

[2] S. Vaudenay, "RFID Privacy based on Public-Key Cryptography," ICISC, M.S. Rhee and B. Lee, eds., Lecture Notes in Computer Science, vol.4296, pp.1–6, Springer, 2006.

[3] A.F. Tenca, G. Todorov, and C.K. Koc, "High-radix design of a scalable modular multiplier," CHES, C.K. Koc, D. Naccache, and C. Paar, eds., Lecture Notes in Computer Science, vol.2162, pp.189–205, Springer, 2001.

[4] H.-K. Son and S.-G. Oh, "Design and implementation of scalable low-power montgomery multiplier," IEEE International Conference on Computer Design (ICCD'04), pp.524–531, IEEE, 2004.

[5] F. Bernard, "Scalable hardware implementating high-radix montgomery multiplication algorithm," Journal of Sustems Architecture, vol.53, pp.117–126, Elsevier, 2007.

[6] A. Ashry and K. Sharaf, "Ultra low power UHF RFID tag in 0.13 μm CMOS," IEEE International Conference on Microelectronics, IEEE 2007.

[7] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," CHES, M. Joye and J.J. Quisquater, eds., Lecture Notes in Computer Science, vol.3156, pp.357–370, Springer, 2004.

[8] Internatonal Organization for Standardization, ISO/IEC 18000-3, Information Technology AIDC Techniques - RFID for Item Management, March 2003.

[9] E.W. Everstine, "Partial key exposure attack on low-exponent RSA," Technial Report, Department of Computer Science, University of Maryland, 2001.

[10] J. Blömer and A. May, "New partial key exposure attacks on RSA," CRYPTO'03, D. Boneh, ed., Lecture Notes in Computer Science, vol.2729, pp.27–43, Springer, 2003.

[11] M. Ernst, E. Jochemsz, A. May, and B. de Weger, "Partial key exposure attacks on RSA up to full Size exponents," EUROCRYPT 2005, R. Cramer, ed., Lecture Notes in Computer Science, vol.3494, pp.371–386, Springer, 2005.

[12] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," CRYPTO 1999, M.J. Wiener, ed., Lecture Notes in Computer Science, vol.1666, pp.388–397, Springer, 1999.

[13] D. Boneh, "Twenty years of attacks on the RSA cryptosystem," Notices of the American mathematical Society (AMS), vol.46, no.2, pp.203–213, 1999.

[14] D. Coppersmith, "Small solutions to polynomial equations, and low exponent RSA vulnerabilities," J. Cryptol., vol.10, pp.233–260, 1997.

[15] D.K. Kim, M.-K. Lee, Y.S. Kang, S.-H. Chung, and W.-J. Yoon, "Design and performance analysis of electronic seal protection systems based on AES," ETRI Journal, vol.29, no.6, pp.755–767, 2007.

[16] H.-M. Sun and M.-E. Wu, "An approach towards rebalanced RSA-CRT with short public exponent," Cryptology ePrint Archive Report, 2005.

[17] J. Sundgren, "Public key algorithms: RSA vs. Elliptic curve cryptography," IdeaByte, May 3, 2002.

Appendix: Availability of Implementation of Pseudo-Random Number Generator on RFID Tag

The proposed protocol uses pseudo-random number generator on RFID tag side as shown in Step 3 of Fig. 1. The implementation of pseudo-random number generator on RFID tag is feasible as shown in Table 2 of [15].