Multiple RFID Tag/Reader Co – Existence Proof Based On Rabin Cryptosystem

Aik Theng Tan[1], Dahlan Abdullah[2], Ronsen Purba[3], Rahmat Budiarto[1]
[1]School of Computer Sciences, Universiti Sains Malaysia, Penang, Malaysia
[2]Computer Center, Universitas Malikussalah, Acheh, Indonesia
[3]STMIK Mikroskil, Medan, Indonesia

Abstract
There are a lot of multiple RFID tag co –existence proofs have been developed by previous researcher such as Yoking Proof [1], On Existence Proof for multiple RFID tag [2] and A Proposed Proof by Thiti et al [3].[1],[2],[3] are applying symmetric secret sharing key method for performing an authentication in between RFID tag and server. In this manuscript, we are going to propose asymmetric secret key algorithm such as Rabin cryptosystem for generating multiple RFID tag co –existence proof. The reasons asymmetric secret key algorithm has been proposed instead of symmetric secret sharing key are:
1) Provides for message authentication: Public key encryption allows the use of digital signatures which enables the recipient of a message to verify that the message is truly from a particular sender [6].
2) Detection of tampering: The use of digital signatures in public key encryption allows the receiver to detect if the message was altered in transit. A digitally signed message cannot be modified without invalidating the signature [6].
For next following sub-chapters in this manuscript [1],[2],[3] and including our proposed method will be shown in more details the way on how they are applied for generating multiple RFID tag /reader co–existence proof and discussion for its security analysis.

Keyword:
Rabin Cryptosystem, Symmetric Secret Sharing Key, Asymmetric Secret Key, RFID – Radio Frequency Identification.

1. Introduction

Rabin Cryptosystem has been developed by Micheal O'Rabin in 1979. The purpose this type of cryptosystem has been developed to enhance the security of encryption data. Rabin cryptosystem is classified as asymmetric secret key and only public key will be revealing out to public. The private keys will be kept in secret and only owner able to unlock an encryption and get back the origin plaintext. It is related to the difficulty of integer factorization. Unlike existing multiple RFID tag co –existence proofs such as [1], [2], [3], it is more on verification hash value and MAC value by applying symmetric secret sharing key approach. By applying Rabin cryptosystem for generating multiple RFID tag co –existence proof, it is not only revealed out robustness security in between symmetric secret sharing key versus asymmetric secret key for breaking an encryption but also to proof Rabin cryptosystem is hard to be attacked if only public key has been known.

From paper [4], [5], there are few sketches of existing network flow have been carried out to identify whether existing multiple RFID tag co–existence proofs network flow such as [1],[2],[3] are able to apply for generating multiple RFID reader co–existence proof. As a result, tag independent each other network flow is identified suitable to apply for generating multiple RFID reader co–existence proof compare to tag dependent each other network flow because it is able to generate multiple RFID tag/reader co–existence proof under different kind of configurations RFID reader and tag. By applying Rabin cryptosystem for generating multiple RFID tag co–existence proof, consideration for above scenario has been taken note to ensure that multiple RFID reader co –existence proof can be generated as well by using same network flow.
Sub-chapters in this manuscript are organized as follow. Section 2 (Yoking Proof), section 3 (On Existence Proof for Multiple RFID Tag), section 4 (A Proof Proposed by Thiti et al), section 5 (Our Proposed Proof) and section 6 (Security Analysis). At last will be the conclusion for this manuscript.

2. Yoking Proof for Multiple RFID Tags

Figure 1 showed that Yoking Proof is applied for generating multiple RFID tag co-existence proof. The descriptions about the proof are as followed:
1) The server sends a random number r to reader.
2) The reader sends the random number r to tag A.
3) Tag A starts to generate a hash value ra applying xa on r and transmitting its back to reader.
4) Reader receives hash value from tag A and sending ra and r to tag B. Tag B generates hash value rb by applying xb on r and applying xb on ra for generating MAC mb.
5) mb and rb are sent to reader by tag B.
6) Reader sends rb to tag A and tag A applies xa on rb for generating MAC ma.
7) Finally, reader computes PAB= (A,B,ma,mb) and submits to server for verification

Manuscript received August 5, 2010
Manuscript revised August 20, 2010

Figure 2 showed that Yoking Proof is applied to proof multiple RFID reader co-existences under 1 RFID tag existences. The descriptions are as below.

1) First, reader 1 and reader 2 each receive a random number r1 and r2 from server.
2) Then, reader 1 and reader 2 transmit r1 and r2 to tag A.
3) Tag A starts to generate hash values for both random numbers, let's called it r1a and r2a. r1a and r2a are generated by applying xa on r1 and r2 respectively.
4) Tag A transmits both of these hash values to reader 1 and reader 2. Reader 1 receives r1a while reader 2 receives r2a.
5) Both of these readers start for looking another tag to generate MAC value.

As a conclusion, we have classified Yoking Proof as symmetric secret key application for generating multiple RFID tag co–existence proof. The server task is to verify hash values and MAC values submitted by RFID reader. All authorized RFID tags secret keys have been computed in server for verification purpose. Beside that, Yoking Proof is classified as tag dependent each other for generating hash value and MAC value protocol. The minimum RFID tag requirements for generating multiple RFID reader co –existence proof as exhibited in Figure 3.

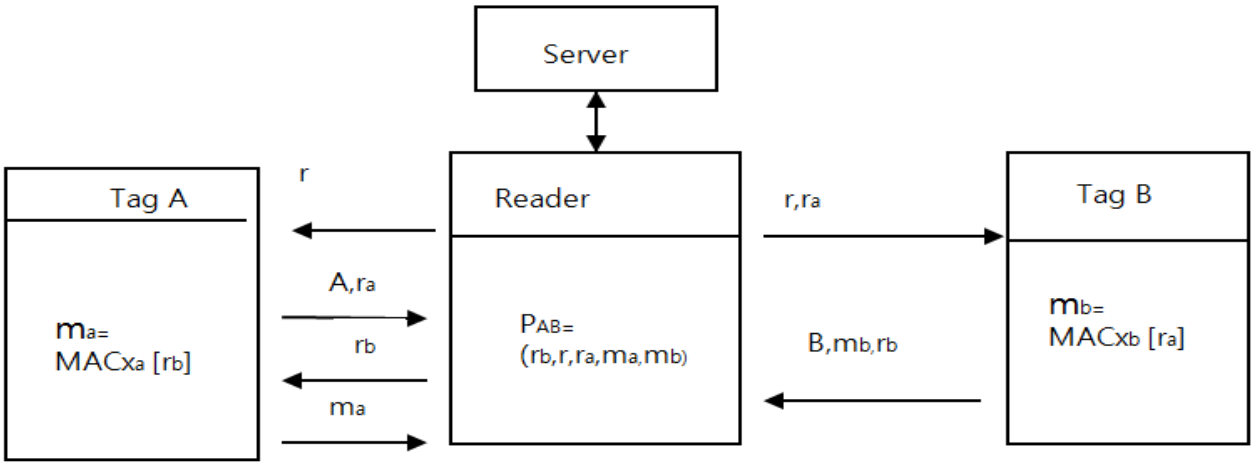

Figure 1: Yoking Proof [1]

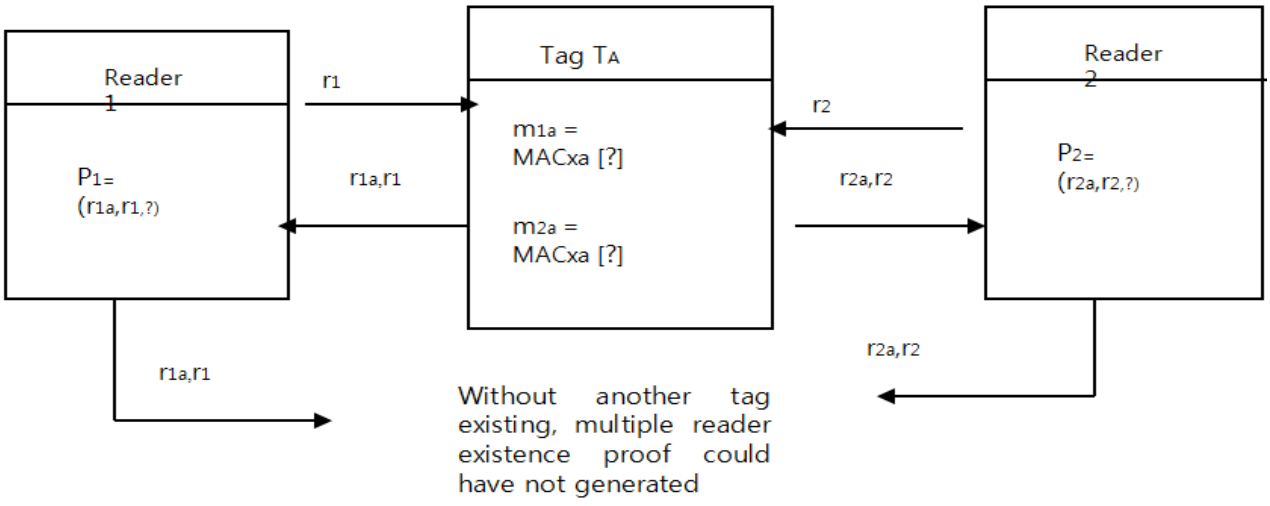

Figure 2: Yoking Proof Applies under two RFID readers in a tag's field [4,5]

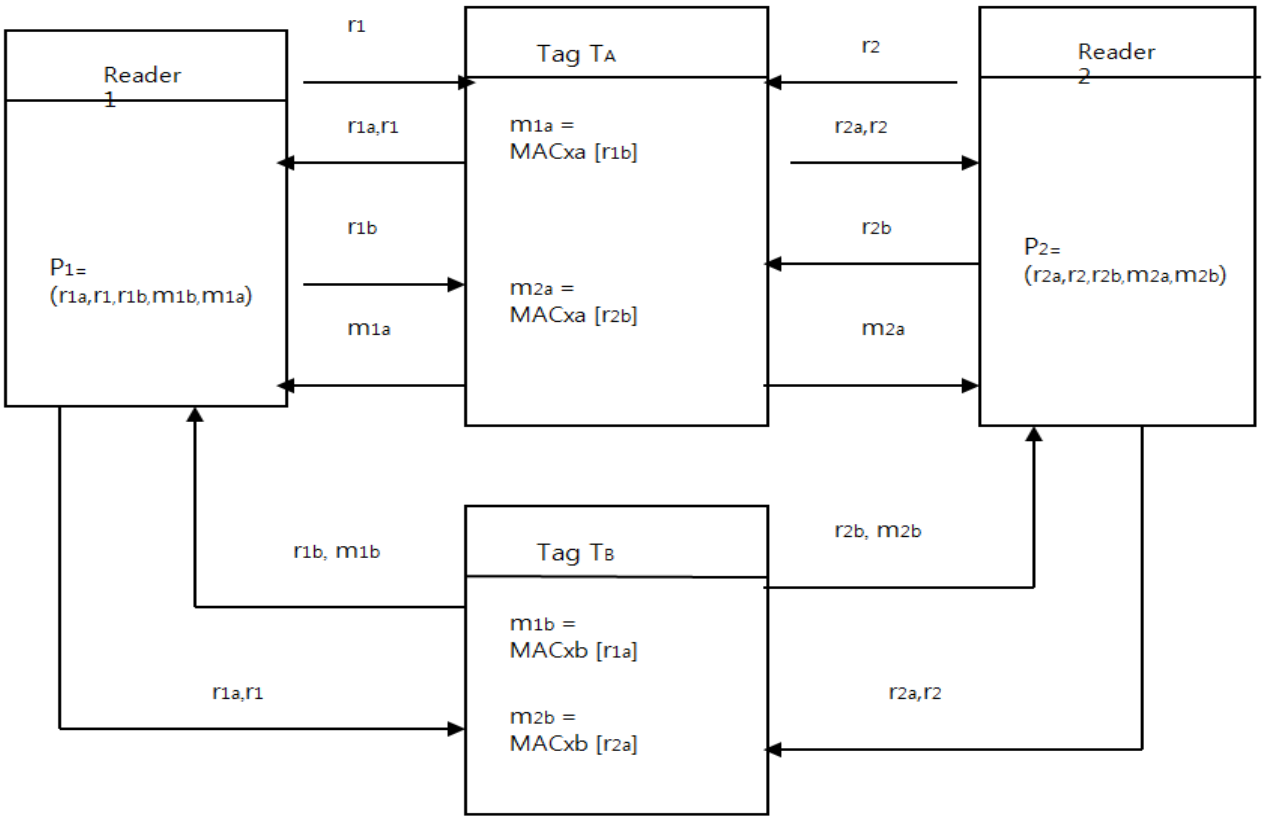

Figure 3: Completed Multiple RFID Reader Co-existence Proof Applying Yoking Proof [4,5]

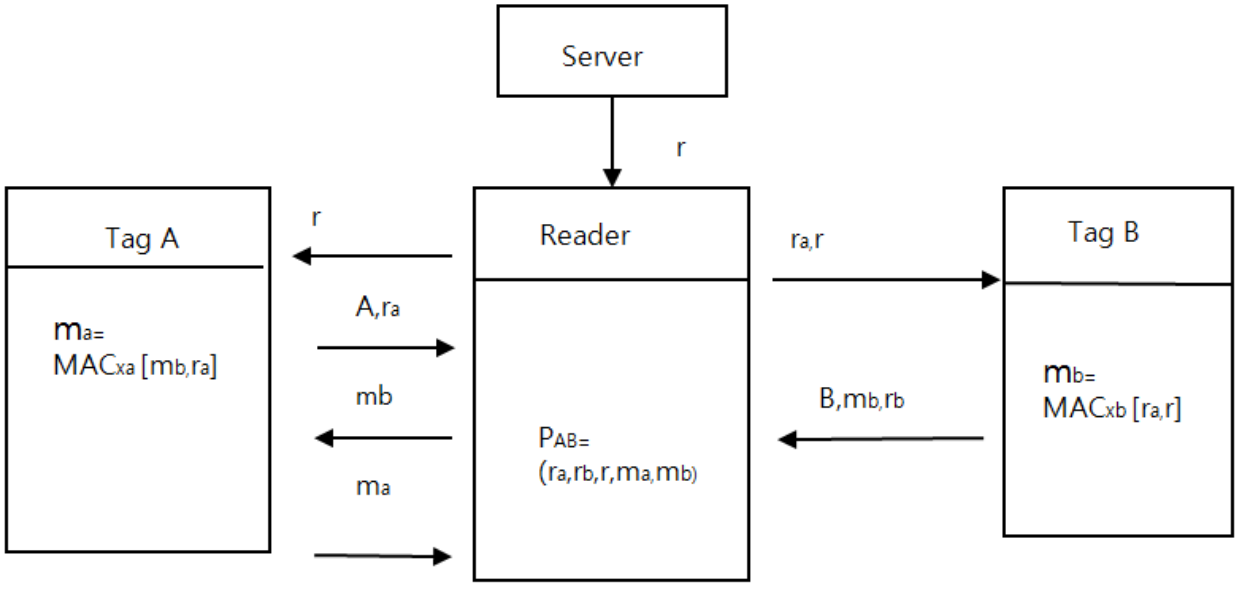

Figure 4: On Existence Proof for Multiple RFID Tags [2]

3. On Existence Proof For Multiple RFID Tags

Figure 4 showed that On Existence Proof is applied to proof multiple RFID tag co-existence. The descriptions about the proof are as followed:

1) The server sends a random number r to reader.
2) The reader sends the random number r to tag A.
3) Tag A starts to generate a hash value ra by applying xa on r and transmitting its back to reader.
4) Reader receives hash value from tag A and ra sends and r to tag B for generating hash value and message authentication code. rb is generated by applying xb on r while message authentication code mb is generated by applying xb on r and ra.
5) mb and rb are sent to reader by tag B.
6) Reader sends mb to tag A and applying xa on mb for generating message authentication code ma
7) Finally, reader computes PAB= (ra,rb,r,ma,mb) and submitting to server for verification.

Figure 5 showed that On Existence Proof is applied to proof multiple RFID reader co-existences under 1 RFID tag existences. The steps are as followed.

1) First, reader 1 and reader 2 each receives random number r1 and r2 from server.
2) Then, reader 1 and reader 2 transmit r1and r2 to tag A.
3) Tag A starts to generate hash values for both random numbers, let's called it r1a and r2a.
4) Tag A transmits both of these hash values to reader 1 and reader 2. Reader 1 receives r1a and reader 2 receives r2a.
5) Both of these readers start for looking another tag to generate a message authentication code by sending (r1,r1a) from reader 1 and (r2,r2a) from reader 2.

To summaries above descriptions, we have classified On Existence Proof as symmetric secret key application for generating multiple RFID tag co–existence proof. The server task is to verify hash values and MAC values submitted by RFID reader. All authorized RFID tags secret keys have been computed in server for verification purpose. Beside that, On Existence Proof is classified as tag dependent each other for generating hash value and MAC value protocol. The minimum RFID tag requirements for generating multiple RFID reader co–existence proof as exhibited in Figure 6.

4. A Proof Proposed by Thiti et al

Figure 7 showed that A Proof proposed by Thiti et al is applied to proof multiple RFID tag co-existence. The descriptions about the proof are as followed:

1) The server sends a random number r to readers.
2) Reader sends the random number r to tag A and tag B.
3) Tag A and Tag B each generates a hash value ra and rb by applying xa and xb on r as a seed.
4) ra and rb are sent to reader by tag A and tag B.
5) Then, the reader generates rt by using ra, rb, and r, based on XOR operation, i.e,. rt=(ra⊕ r ⊕ rb) [3].
6) The reader sends rt to tag A, and tag A reacts by generating and sending the MAC ma, applying xa on ra and rt, to the reader [3].
7) The reader sends rt to tag B, tag B reacts by generating and sending the MAC mb, applying xb on rb and rt, to the reader [3].
8) The reader generates the MAC mr, applying xr on r and computes PAB= [ra,rb,rt,ma,mb,mr] [3].
9) The reader sends PAB to the server for verification.

Figure 8 showed that A Proof proposed by Thiti et al is applied to proof multiple RFID reader co -existence under 1 RFID tag existences. The steps are as followed:

1) The server sends random numbers r1 and r2 to reader 1 and reader 2.
2) Reader 1 and Reader 2 send the random numbers to tag A.
3) Tag A generates two hash values (ra, ra') by applying xa on r1 and r2.
4) Hash values are transmitted back to readers by tag A.
5) Then, the readers generate rt and rt' by using ra, ra', r1, and r2, based on an XOR operation, i.e, rt=(ra⊕r1) (Reader 1) , rt'=(ra'⊕ r2) (Reader 2).
6) Readers send rt and rt' to tag A, tag A reacts by generating and sending the MAC ma and MAC ma' to readers, applying xa on ra and rt, (Reader 1) applying xa on ra' and rt' (Reader 2) [4].
7) Reader 1 generates the MAC mr by applying xr on r1 while reader 2 generates MAC mr' by applying xr' on r2. Reader 1 computes P1= (ra,r1,ma,mr,rt) and Reader 2 computes P2= (ra',r2,ma',mr',rt').
8) The readers send P1 and P2 to the server for verification.

It is classified A Proof proposed by Thiti et al as symmetric secret key application for generating multiple RFID tag co–existence proof. The server task is to verify hash values and MAC values submitted by RFID reader. All authorized RFID tags secret keys have been computed in server for verification purpose. Beside that, An Improve Proof is classified as tag independent each other for generating hash value and MAC value protocol. The minimum RFID tag requirements for generating multiple RFID reader co–existence proof is one therefore it is ideal for application in generating multiple RFID tag or reader co –existence proof for different kind of configurations.

5. Our Proposed Proof

Figure 9 showed that multiple RFID tag co –existence proof has been generated by applying asymmetric secret key such as Rabin cryptosystem for authenticating the random number has been sent out by server. It is quite different from existing proofs such as [1],[2],[3] because an algorithm for Rabin cryptosystem is more on numerical factorization rather than verification of hash values and message authentication code. Below are the steps exhibited on how multiple RFID tag co –existence proof can be generated as well by using Rabin cryptosystem.

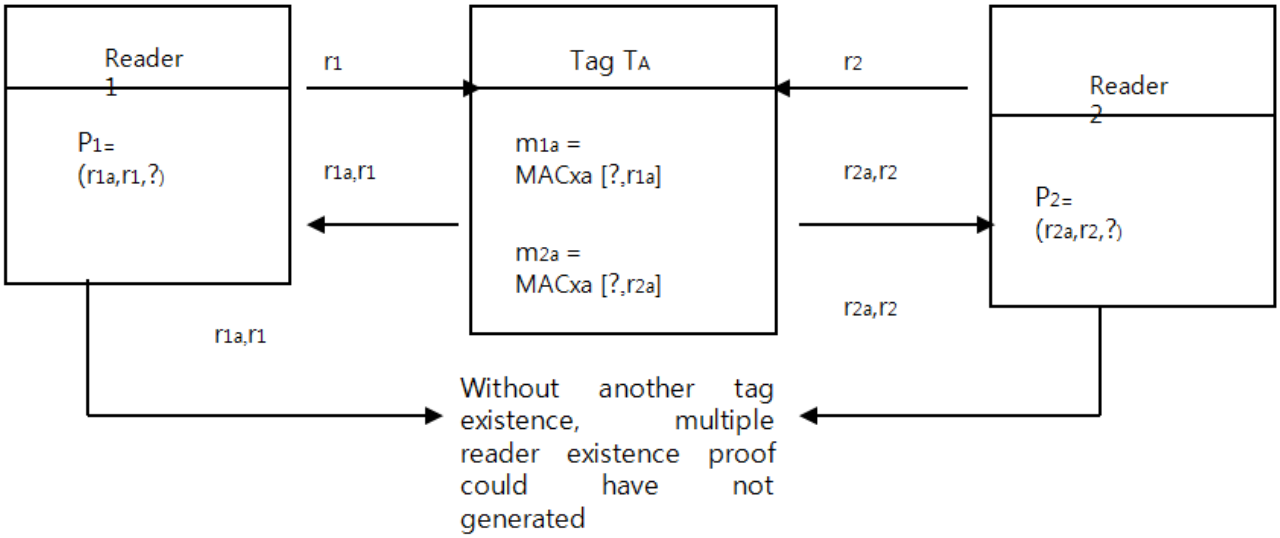

Figure 5: On Existence Proof Applies under Two RFID Readers in a Tag's Field [4,5]

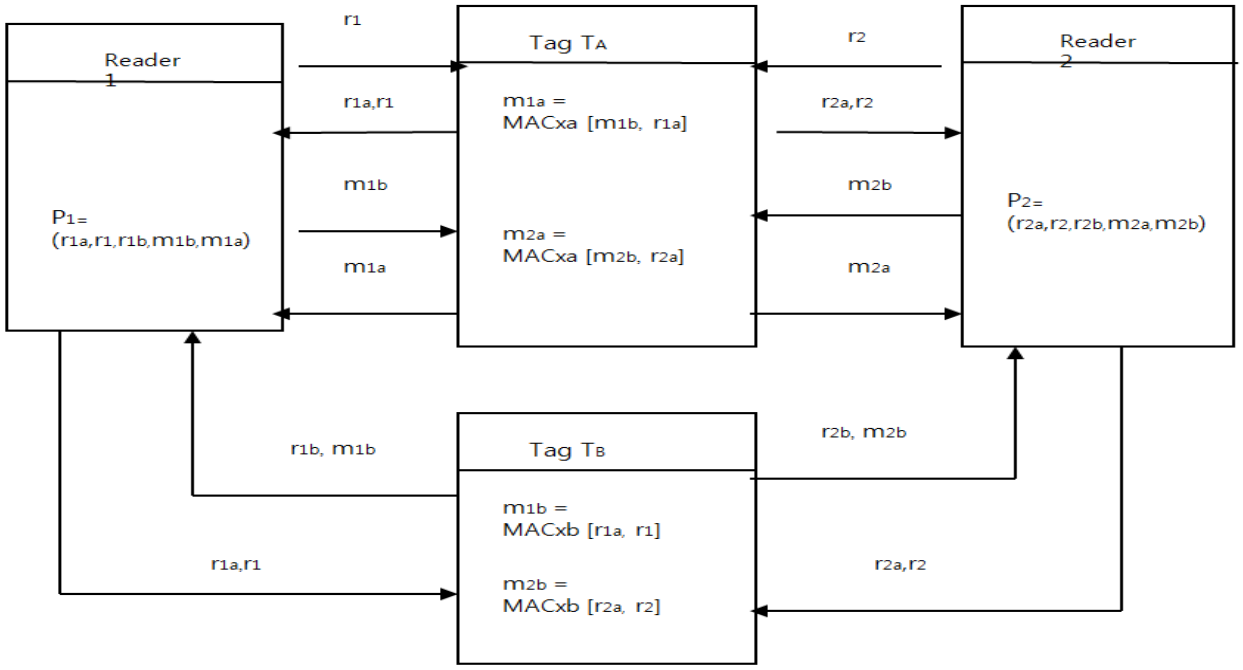

Figure 6: Completed Multiple RFID Reader Co-existence Proof Applying On Existence Proof [4,5]

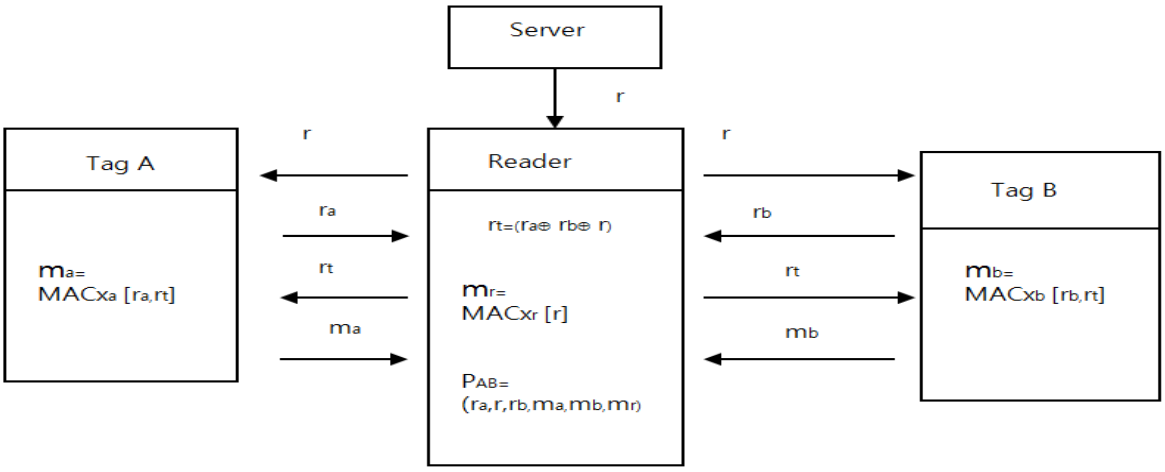

Figure 7: A Proof Proposed by Thiti et al [3]

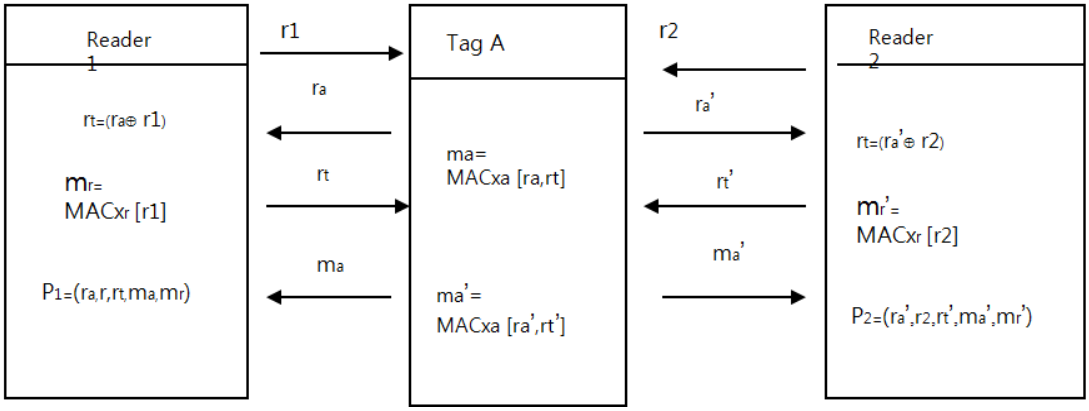

Figure 8: Multiple RFID Reader Co-existence Proof (A Proof Proposed by Thiti et al)

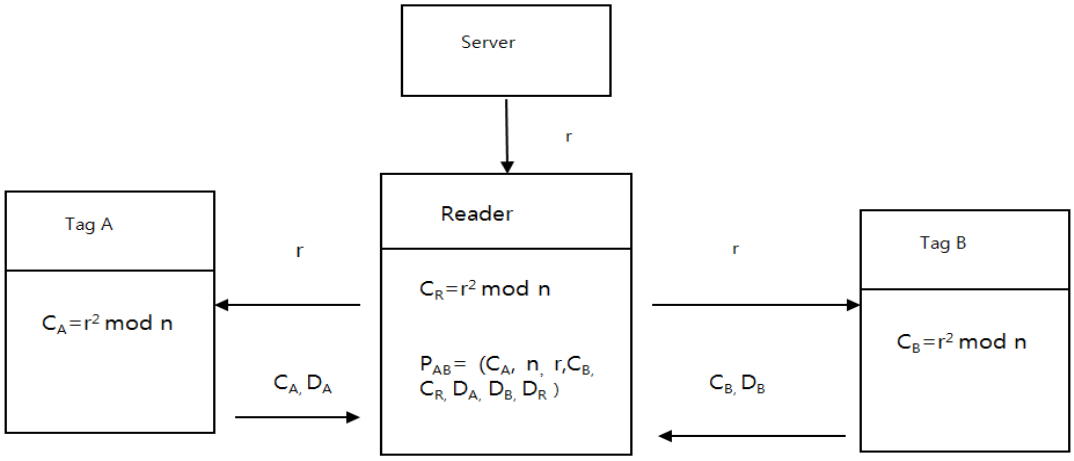

Figure 9: Multiple RFID Tag Co –Existence Proof Based On Rabin Cryptosystem

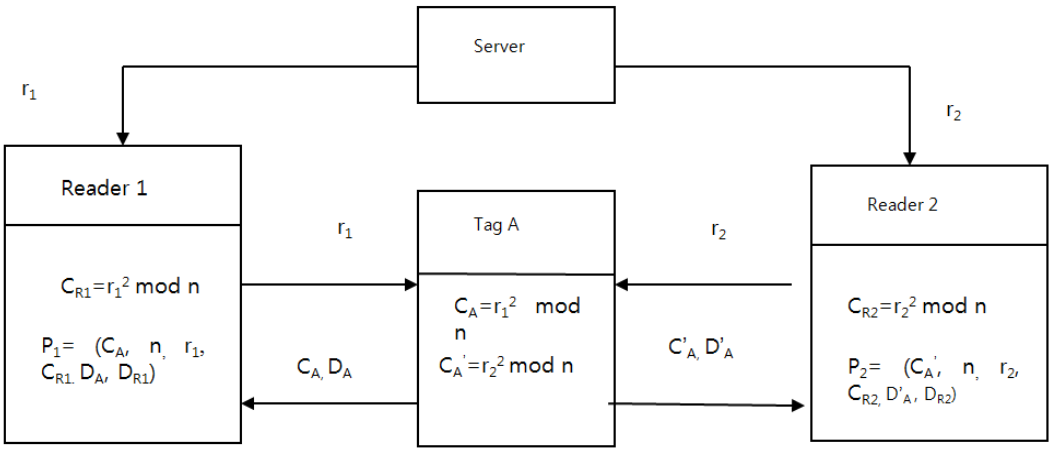

Figure 10: Multiple RFID Reader Co –Existence Proof Based On Rabin Cryptosystem

1) Initially, server (n) and RFID tag / reader (Pt, Pr) exchange public key as a key agreement protocol.
2) The server sends a random number r to reader.
3) RFID reader starts to generate its cipher text and digital signature (CR , DR).
4) The reader sends the random number r to tag A and tag B.
5) Each of RFID tag starts to generate a cipher text (CA&CB) and provides its digital signature before returning to RFID reader.
6) RFID reader receives cipher text (CA&CB) and digital signature (DA & DB) from each RFID tag before sending to server for performing verification.
7) After RFID reader has received all parameters as mentioned in step 4, server will apply public keys have been received from RFID tag and reader for verification digital signature while applying its private key for performing decryption.
8) Below is an algorithm for decryption cipher text by applying Rabin Cryptosystem.
a and b satisfying a*p + b*q = 1, using the extended Greatest Common Divisor algorithm, computed once when the keys are generated [7].

$r = c^{(p+1)/4} \bmod p$ (1)

$s = c^{(q+1)/4} \bmod q$ (2)

$x = (a*p*s + b*q*r) \bmod n$ (3)

$y = (a*p*s - b*q*r) \bmod n$ (4)

9) There will four plaintexts appear after applying steps above and one of them is an original plaintext.

Figure 10 exhibited that multiple RFID reader co – existence proof could be generated as well by using Rabin cryptosystem. The steps are below:
1) Initially, server (n) and RFID tag / reader (Pt, Pr) exchange public key as a key agreement protocol.
2) The server sends random numbers r1 and r2 to reader 1 and reader 2.
3) Reader 1 and reader 2 generate its cipher text and digital signature (CR1, CR2) (DR1, DR2).
4) Reader 1 and Reader 2 send these random numbers to tag A.
5) Tag A generates an encryption to these 2 random numbers by applying public key sent out by server.
6) Cipher text (CA &C'A) and digital signature (DA & D'A) are transmitted back to readers from tag A.
7) Different random numbers produce different cipher text even though same public key has been applied for encrypting the plaintext.
8) In order to generate an origin random number sent out by server, each of the reader will submit cipher text and digital signature collect from RFID tags for performing decryption and verification digital signature. Server will apply public key sent out by each of RFID tag/reader for performing verification digital signature while applying its private key for decrypting cipher text.

From above descriptions, our proposed method is classified as asymmetric secret key application for generating multiple RFID tag or reader co –existence proof. Each of RFID tag private keys will be stored in each of particular RFID tag without reveal out to public meanwhile public key receives from server will be applied for performing encryption data. Rabin algorithm is more ideal for this application because it is deal with the difficulty to get back two origin prime numbers from product of two prime numbers. Beside that, decryption cipher text by Rabin algorithm will generate back the origin random number sent out by server rather than just verification hash values and message authentication code values as [1], [2], [3].

6. Security Analysis

In order to proof that asymmetric secret key algorithm such as Rabin cryptosystem can be applied for generating multiple RFID tag/reader co –existence proof, we will provide three main scenarios to prove that application of asymmetric secret key is much more secured than symmetric secret sharing secret key. The reasons are as followed:

1) Secure channel is required for secret key exchange – There are a necessary for sender and receiver to have a secure channel for exchange secret key in symmetric mode if not there will be a possibility for adversary to eavesdrop the secret key and become fake server in this scenario [6].

2) Detection of tampering – By using public key encryption, digital signature is required from sender. Hence, receiver able detect if message is altered during transit because digitally signed message cannot be modified without invalidating the signature. Digital signature enable server to detect random number sends to RFID tag for encryption whether is tampered by others or not during transit in this scenario [6].

3) Provides for message authentication – Public key allows use of digital signature. Hence, recipient manages to detect the message truly from a particular sender. Digital signature enable server to detect fake RFID tag existences in this scenario because RFID tag applies public key sent by server for encryption and need to include digital signature as well before can transmit the encryption for verification [6].

A small example numerical factorization calculation to show that random number sent out by server will be able to recover back by applying Rabin algorithm. Following is an example:

Server chooses two primes p and q each equal to 3 modulo 4, and forms the product n = p*q whereby n is a public key while p & q are private keys. Hence, server will never reveal out its private keys to others same as well for each of RFID tag. Initially, server will send out public key n and random number (r) in plaintext form for generating cipher text. Take two private keys as an example p=7, q=11, hence public key will be n=7*11=77. Each of RFID tag or reader will apply Rabin encryption algorithm where the formula is c = r2 mod n for generating cipher text (C). Let random number in plain text form (r). Assume that message will be represented by value 45. Then, cipher text generates will be c = 452 mod 77=23.

After cipher text has been generated by RFID tag/reader, it will compute in the cipher text in digital signature by applying its private key as a signature and send it to server for verification. Server will verify digital signature receive from each of RFID tag/reader by applying public key sent out by each of RFID tag/reader. Then, server will apply its private key for performing decryption cipher text. Below are the 6 equations for recovering back the plain text by applying Rabin algorithm.

$$r = c^{(p+1)/4} \bmod p \quad (1)$$
$$s = c^{(q+1)/4} \bmod q \quad (2)$$
$$m_1 = (a*p*s + b*q*r) \bmod n \quad (3)$$
$$m_2 = (a*p*s - b*q*r) \bmod n \quad (4)$$
$$m_3 = n - m_1 \quad (5)$$
$$m_4 = n - m_2 \quad (6)$$

In order to get value a and b from equations 3 and 4, a and b must satisfy equation a*p + b*q = 1, using the extended Greatest Common Divisor algorithm

After decryption r=4, s=1, m1=67, m2= 45, m3 = 10, m4 = 32

7. Conclusion

Based on sketches in Figure 9 and Figure 10, multiple RFID tag or reader co–existence proof can apply Rabin algorithm to generate as well besides applying symmetric secret sharing key.

As a conclusion for this paper, we have proposed that asymmetric secret key can be applied as well for generating multiple RFID tag or reader co -existence proof. A comparison has been made in between symmetric sharing secret key and asymmetric secret key due to its security for generating multiple RFID tag/reader co –existence proof. There is a high risk for secret key to be eavesdropped by adversary upon key agreement protocol has been performed by sender and receiver because it really needs a very secure channel. If adversary successfully eavesdrop the secret key, it would able to communicate with server and act as an authorized RFID tag. Unlike asymmetric secret key, digital signature is required from every single RFID tag who applies a public key Ps sent out by server. Hence, server able recognizes an unauthorized RFID tag existences. Besides that, digital signature helps to identify if there is a tempering data's scenario occurs and helps to resolve replay attack where existence multiple RFID tag co – existence proofs confront it.

Notation:

r, r_1, r_2 – Random Numbers
r_a, r_b, r_a', r_b', r_{1a}, r_{2b} – Hash Values
P_{AB} – Proof that Tag A and B scanned simultaneously
x_a, x_b,x_r – Secret Key sharing in between tag and server
m_a,m_b,m_r,m_a',m_{1a},m_{2b},mr-Message Authentication Codes
P_1,P_2 – Proof that Reader 1 and 2 read the tag simultaneously

References

[1] A. Juels. "Yoking Proofs" for RFID Tags," in *Proc. 1st International Workshop on Pervasive Computing and Communication Security*. IEEE Press.2004

[2] S. Piramuthu. "On Existence Proofs for Multiple RFID Tags," *Proc. Of IEEE International Conference on Pervasive Services* (ICPS'06), pp. 26-29, 2006.

[3] Thiti Nuamcherm, Piya Kovintavewat, Charturong Tanibundhit, Urachada Ketprom, Chaichana Mitrpant."An Improved Proof for RFID Tags" http://home.npru.ac.th/piya/Documents/Thiti_ECTI20 08.pdf

[4] Aik Theng Tan, Rahmat Budiarto, "Applying Multiple RFID Tag Co-existence Proof For Generating Multiple Reader Existence,"*Proceedings 2009 IEEE International Conference on Antennas, Propagation and Systems (INAS 2009)*, Johor Baru, Malaysia, Dec. 3-5 2009.

[5] Aik Theng Tan, Rahmat Budiarto, "An Analysis Of Multiple Tag Co-Existence Method For Generating Multiple Reader Co-Existence Proof," *Journal of Communication and Computer, ISSN 1548-7709*, Vol 7 Issue 4, 2010.

[6] A Comparison Of Symmetric Key and Asymmetric Key Encryption Methods http://webupon.com/security/a-comparison-of-symmetric-key-and-asymmetric-key-encryption-methods/

[7] Neal R.Wagner "The Laws of Cryptography: Rabin's Version Of RSA" http://www.cs.utsa.edu/~wagner/laws/Rabin.html

Aik Theng Tan received his B.ENG in Electrical & Electronics Engineering from Northumbria University in 2007. Currently, he is a Master of Science student in USM. His research interests include cryptography and network security.

Rahmat Budiarto received B.Sc degree from Bandung Institute of Technology in 1986, M.Eng, and Dr. Eng in Computer Science from Nagoya Institute of Technology in 1995 and 1998, respectively. Currently, he is an Associate Professor at School of Computer Sciences Universiti Sains Malaysia. His research interest includes IPv6, network security and Intelligent Systems. He is a member of IEEE Computer Society.

Dahlan Abdullah, received his B.Sc. degree in Informatics Engineering from Indonesian Islamic University, Yogyakarta-Indonesia in 1999. Currently, he is a Master of Science student in USM. His Research interest is Computer Network. Currently he is Chief of The Computer Centre, Universitas Malikussaleh, Aceh Indonesia. He is an active member of INHERENT (Indonesian Higher Education Network), JARDIKNAS (National Education Network), and APTIKOM (Association of Computer and Information University).

Ronsen Purba, received his B.Sc. degree in Mathematics from Universitas Sumatra Utara in 1985, and M.Sc. in Computer Science from Indiana University, Blomington, USA, in 1991. Currently he is the head of Research & Community Services Division at STMIK Mikroskil. His research interests include Data Security, Cryptography, and knowledge-based Systems.