ISSN:0975-9646

A Survey on Implementing Public Key Cryptography on RFID Tags

Prajnya Priyadarsini Satapathy, Debi Prasad Mishra

Department of Information Technology, College of Engineering and Technology, Bhubaneswar,Odisha, India

***Abstract*—Radio Frequency Identification (RFID) is an emerging technology, which brings enormous productivity benefits in applications, where objects have to be identified automatically. This paper presents issues concerning security and privacy of RFID systems. RFID tags are a new generation of barcodes with added functionality. The risk related to using the tags for several applications. In particular if no appropriate cryptography measures are taken, the privacy of a user carrying tagged items can be severely damaged. To enable these applications and at the same time minimize the risks, Public Key Cryptography (PKC) offers attractive solution. We focus on the problem of anti-counterfeiting measures that can be provided by RFID tags. PKC –based identification protocols are useful for this application.**

***Keywords*— RFID, Authentication, public key cryptography, WIPR**

I. INTRODUCTION

Public key cryptography is a class of cryptographic algorithm which requires two separate keys i.e. private key and public key. Two pair of this key is linked mathematically, where public key is used to encrypt plain text or to verify a digital signature and private key is used to decrypt the cipher text or to create digital signature. The problem of anti-counterfeiting measures is focussed such that it can be provided by RFID tags. Counterfeit products are fake replicas of the real product. Counterfeit consumer products have a reputation for being quality and may even include toxic elements. This has resulted in the death of hundreds of people due to automobile and aviation accidents, poising, ceasing to take essential accidents.

Increase of counterfeit goods translates into a large source of losses for manufactures. The following numbers will provide an idea of the extent and criticality of the problem: (i) it has been estimated that the world market for counterfeit goods was worth between 350 and 385 billion USD in 2001 and it was expected to surpass the 500 billion USD per year mark by 2004 [2], (ii) in the copyright industry, almost 50% of all motion picture videos, more than 40% of all business software, and a third of all music recordings are pirated copies, (iii) about 10% of clothing, fashion and sportswear are fake and the online sales of luxury good searches 25 billion USD annually, (iv) in the automotive industry 5% to10% of all spare parts are counterfeits, and (v) between 5% and 8% of the 500 billion USD in medicines sold worldwide are counterfeit as estimated by the World Health Organization , in developing countries the percentage of counterfeit drugs account for up to 60% of all drugs [3,4].It is noticed that the above points only point to the economical consequences of counterfeit products. However, in the particular case of the pharmacy industry, counterfeit products have a direct (negative) impact on the health and life of thousands of people worldwide. It is clear that new technologies need to be put in place to thwart the counterfeiting threat. RFID has been identified as one of these technologies as shown for example by legislation introduced in the US mandating use of RFID technology as anti-counterfeiting technology for at-risk pharmaceuticals for all medicines in the supply chain by the end of 2010 [4]. RFID-tags that withstand general cloning attacks (including physical ones) are introduced. Based on an Integrated PUF (I-PUF) [7, 8, 9] a PUF-Certificate-Identity Based identification scheme was introduced. This scheme allows for off-line authentication.

II. BACKGROUND

The electronic product code (EPC) system is one of the world's most ambitious pervasive computing projects. It aims to replace today's familiar 14-digit optical-scan universal product code bar codes with radio-frequency identification(RFID) tags operating in the ultra-high frequency (UHF) band, which are based on the EPC standard [5]. As noted in [6], the additional capabilities of EPC tags create considerable privacy issues which did not exist with optical barcodes.

A. Public key cryptography(PKC)

Public-key cryptography is used as a method of assuring the confidentiality, authenticity and non reputability of electronic communications and data storage. Public-key cryptography and related standards and techniques underlie security features of many Red Hat products, including signed and encrypted email, form signing, object signing, single sign-on, and the Secure Sockets Layer (SSL) protocol. This document introduces the basic concepts of public-key cryptography.

Encryption is the conversion of data into a form called a cipher text, which cannot be easily understood by unauthorized people. Decryption is the process of converting encrypted data back into its original form. A computer can be used in an attempt to break the cipher. Due to encryption/decryption wireless circuits are easier to tap than their hard-wired counterparts. In order to easily recover the contents of an encrypted signal, the correct decryption key is required .The key is an algorithm that encodes the work of encryption algorithm.

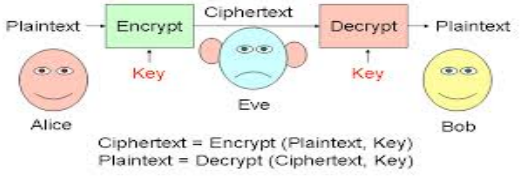

Figure 1 Public key cryptography

III. RELATED WORK

A first set of related papers to ours are [11] and [12]. Both deal with the cloning problem of RFID-tags and hence with the problem of using RFID-tags for anti-counterfeiting purposes. The focus of these papers is on efficient protocols for authenticating these tags. In these papers, one focuses on authentication of RFID-tags in the on-line situation; i.e. when the reader shares a secret with the RFID-tag that is being authenticated. Clearly, for applications with large deployments of RFID tags, this is not a reasonable assumption.

IV. CHALLENGES OF RFID

A. RFID Protocols To Protect Against Cloning Attacks

RFID authentication protocol is presented that enforces user privacy and protects against tag cloning. RFID tags are often envisioned as a replacement for barcodes, having a number of important advantages over the older bar-code technology. Apart from their small size which allows them to be implanted within objects, identification by frequency allows objects to be read in large numbers without the need for a visual contact. Furthermore, RFID identifiers are long enough so that every object has a unique code. Such universal uniqueness means that a product may be tracked as it moves from location to location finally ending up in the consumer's hands.

In addition to the privacy risks imposed by the incorrect use of RFID tags, RFID deployment may suffer from tag cloning. As a motivating example [3], consider an attacker that spoofs a valid tag in an attempt to remove an item without being noticed. If the attacker can install a replacement tag which can continue authenticating itself to a reader, then the attacker can fool the system into believing the product is still on the shelf. Alternatively, an attacker can replace tags for expensive items with tags for cheaper ones.

Contribution on this work is twofold: First a light weight protocol is proposed that can be used for authenticating the tags while at the same time avoiding traceability. The protocol allows for both tag-to-reader and reader-to-tag authentication. The first form of authentication is needed in order to prevent the tag cloning attack mentioned above while the second is needed in order to prevent queries by unauthorized readers which can be used in violating user privacy.[27]

Second, protocol against a multitude of attacks are analysed that seem important in designing new RFID protocols and we identify mutual authentication as one of the key challenges in the area. Tags must reveal their identities only to authorized readers but this should happen only if the reader has been authenticated to the tag.[27]

V. STUDY OF VARIOUS IMPLEMENTATION PLATFROMS

Use of RFID as an anti-counterfeiting technology is at present rather primitive. The whole security relies on the premise that an RFID tag is harder to copy than a bar code. Although, this is certainly true, it will only be a matter of time until counterfeiters can clone simple RFID tags.[1] The anti-counterfeiting problem can also be rephrased as an authentication problem. RFID-tags contain some secret reference information that is used to check their authenticity. To avoid counterfeiting RFID-tags have to be unclonable. First, this implies that it should be hard to make a physical clone. Secondly, this also means that retrieving the secret reference information by attacking the protocols that are carried out between the reader and a tag (proving its authenticity) should be unfeasible. Protection against physical unclonability is provided by using physical countermeasures such as physical unclonable functions and protection against active or passive attacks on the protocols is provided by cryptographic techniques such as digital signatures and secure identification protocols. [1]

RFID-based identification is an example of an emerging technology which requires authentication as a cryptographic service [18]. This property can be achieved by symmetric as well as asymmetric primitives. Previously known work considered only symmetric-key algorithms e.g. AES [19]. The suitability of Public-Key (PK) algorithms for RFID is an open research problem as limitations in cost, area and power are quite severe.

A. Advanced Encryption Standard(AES)

Advanced Encryption Standard (AES) is the current standard for secret key encryption.AES was created by two Belgian cryptographers, *Vincent Rijmen* and *Joan Daemen*, replacing the old Data Encryption Standard (DES). The Federal Information Processing Standard 197 used a standardized version of the algorithm called *Rijndael* for the Advanced Encryption Standard. The algorithm uses a combination of Exclusive-OR operations (XOR), octet substitution with an S-box, row and column rotations, and a Mix Column.

AES (Advanced Encryption System) allows for block sizes of 128, 168, 192, 224, and 256 bits. AES allows key sizes of 128, 192, and 256 bits [15]. The standard encryption uses AES-128 where both the block and key size are 128 bits. The block size is commonly denoted as N_b and the key size is commonly denoted as N_k . N_b refers to the number of columns in the block where each row in the column consists of four cells of 8 bytes each for AES-128 [16]. The following example will show how data is broken up into blocks. Using AES-128 means that each block will consist of 128 bits. Nb can be calculated by dividing 128 by 32. The 32 comes from the number of bytes in each column. In this case, Nb is 4. The original plain text is stored in bytes in a block. For example, the text "This is a test…" will be stored in a block as shown below in Figure 2.

Block

	0	1	2	3
0	T		a	s
1	h	i		t
2	i	s	t	.
3	s		e	.

Figure 2 AES-128 block example

Each character is stored in a cell of the block. The blank cells shown in the diagram are not really blank as they represent the spaces in the text. Depending on how the algorithm is implemented the characters may be stored as integer values, hexadecimal values, or even binary strings. All three ways represent the same data. Most diagrams show the hexadecimal values, however integer and string manipulation is much easier to do when actually programming AES. Figure 1 shows the values as characters for demonstration purposes to show how the text is stored into the block. The plain text is stored into blocks column by column and block by block until all the data is stored [16]. In the example used above there were exactly 16 characters used for simplicity. In order to use the *Rijndael* algorithm the data must be a multiple of the block size, since all blocks need to be complete. When the data is not a multiple of the block size some form of padding must be used. Padding is when extra bits are added to the original data. One forms of padding includes adding the same bytes until the desired size is reached. Another option is padding with all zeros and having the last byte represent the number of zeros. Padding with null characters or random characters are also forms of padding that can be used. [17] Once a form of padding is chosen the data is represented as some number of complete blocks. The last thing needed before using the algorithm is the key. The key also known as the cipher key is also the same size as the block in this example. Unlike most data and transformations the cipher key can have any values chosen by the designer with no restrictions as long as the key is the correct length. The key is also stored as a block similar to the plain text. [16] When the plain text data is stored into blocks and the key is chosen the *Rijndael* encryption algorithm can be applied.

B. *Public key (PK- based) Protocol*

i. Protocols are present where readers are online, share a secret with the tags. Protocols for this case are very cheap and can easily implemented on a high-end RFID tag. By today's standard, the tags that would correspond to a mid to high range tags are considerable paragraphs must be indented. Although, it is anticipated that in the near future price pressure will continue to limit the number of gates in the ultra low cost tags, it can also be envisioned that eventually this number of gates will be available on all tags. The cost of a security solution is directly dependent on the thing(s) which are being emphasized and safeguarded. Thus, just as there are applications for which the solutions would be too expensive. There are also RFID applications for which such cost might be acceptable

ii. When the readers are off-line verification they do not share any secret with the tag. The protocols investigated were only secure against passive attacks and the efficiency of protocols (Okamoto-identification protocol) secure against active and concurrent attacks. It is shown that only a small price for much additional security has to be paid.

C. WIPR Encryption Scheme

The WIPR is a variant of the Rabin's encryption scheme presented in [22], first discussed in [23], which is provably as secure as factoring large numbers. In Rabin's scheme, the private key consists of two large prime numbers p and q. These are multiplied to form the public key $n = p \cdot q$. The plaintext P is typically generated from a shorter string (in our case an ID) by padding it with random bits until it is as long as n. To encrypt a plaintext P in this scheme, the sender calculates the cipher text M as its square, reduced modulo n:

$M = P2 \pmod n$

To decrypt a cipher text, the receiver calculates the square roots of M modulo p and q, and then combines the resulting values using the Chinese Remainder Theorem [§2.4.3].

Each cipher text has two possible roots modulo p and two roots modulo q ($\pm m \pmod p$ and $\pm m \pmod q$), leading to four possible plaintexts for each cipher text. To allow the receiver to determine which of the four possible plaintexts the correct one is, the sender typically adds some redundancy to the message. The encryption element of Rabin's scheme is relatively easy to implement, requiring only a single multiplication and modular reduction. However, modular reduction is a RAM-intensive process, a fact that limits the applicability of Rabin's algorithm to low-resource devices such as smart cards. To reduce the resource requirements of Rabin's scheme, *Naccache* in [24] and *Shamir* in [25] and later [26] suggested a RAM efficient variant, replacing the modular reduction step by an addition of a large random multiple of n, where the size of the random value r is at least 80 bits longer than the size of n (to have no detrimental effects on security): $M = P2 + r \cdot n$ The decryption algorithm is precisely identical to Rabin's original scheme. Shamir proved that the security of this resource-reduced scheme and the original Rabin scheme are equivalent. The reduced scheme is easier to implement since it has only multiplication operations and not modular reductions. In terms of space requirement, the problem of storing P2 was replaced by the challenge of storing the large random number r. However, since r is written to only once per protocol execution [25], suggested that it should be stored in EEPROM, which is plentiful on smart cards, and not on the more scarce RAM. However, rewritable EEPROM is cheap on smart cards and prohibitively expensive on RFID tags, due to the high power cost of the write operation.

The final resource reduction in the Rabin scheme was presented in the WIPR scheme [21]. WIPR replace sr with the output of allow-resource reversible stream cipher. This cipher is implemented by creating a Feistel structure [28], a well known cryptographic construct used in symmetric ciphers such as DES and TEA. To make use of this cryptographic building block to provide secure identification, a challenge response construction was used, adding a reader-supplied random challenge to the plaintext P.

Protocol steps:

Given the above description, following is an outline of the protocol steps:

1. Setup: The tag is provided with the public key n and a signed unique identifier I D. The reader is provided with the private key (p,q).
2. Boot: The reader generates a random bit string R_r , where $|R_r| = \alpha$. The tag generates two random bit strings R_{t1} and R_{t2} , where $|R_{t1}| = |n|-\alpha-|I\ D|and|R_{t2}| = |n|+\beta$. and α, β are security parameters (both set to 80 in our implementation).
3. Challenge: The reader sends R_r to the tag.
4. Response: The tag generates a plaintext as follows: P =Rr#Rt,1#I D, where # denotes concatenation, and then transmits the following message: $M = P_2 + R_{t2} \cdot n$
5. Verification: The reader uses the private key to decrypt M. There are four candidate decryptions, so the reader checks which of the four possible decryptions contain the value of the challenge R_r it sent to the tag. If such a plaintext is found, the reader outputs the value of I D. In all other cases, the authentication fails.

The WIPR protocol is based on public-key cryptography—the public key stored on the tag allows messages to be encrypted, but does not allow messages to be decrypted, even if those messages were previously transmitted by the same tag. In contrast, a system based on secret-key cryptography must use the same key both on the reader and on the tag, entire document should be in Times New Roman or Times font. Type 3 fonts must not be used. Other font types may be used if needed for special purposes.

VI. CONCLUSION

Passive radio-frequency identification (RFID) tags have long been thought to be too weak to implement public-key cryptography: It is commonly assumed that the power consumption, gate count and computation time of full strength encryption exceed the capabilities of RFID tags. Two low-resource implementations of a 1,024-bit Rabin encryption variant called WIPR—in embedded software and in hardware. An optimized WIPR implementation is presented which is small enough to fit on an RFID tag: Using a variety of hardware design optimization techniques, a working point is identified that is well within a tag's power and area budgets, and is fast enough for the intended application. It is concluded that the public-key approach is a viable design alternative for supply-chain RFID EPC tags.

REFERENCES

[1] Batina, Lejla, et al. "Public-key cryptography for RFID-tags." *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on*. IEEE, 2007.

[2] ICC Policy Statement: The fight against piracy and counterfeiting of intellectual property. Submitted to the 35th World Congress, Marrakech, Document no450/986, ICC, June 1st, 2004.

[3] Intellectual Property: Source of innovation, creativity, growth and progress. Technical report, ICC, August 2005.

[4] R. Koh, E. W. Schuster, I. Chackrabarti, and A. Bellman. Securing the Pharmaceutical Supply Chain. White Paper MIT-AUTOID-WH-021, Auto-Id Center MIT, Cambridge, Ma 02139-4307, USA, September 1st, 2003. Available at http://www.mitdatacenter.org/MIT-AUTOIDWH021.pdf.

[5] Epcglobal inc.: EPC radio-frequency identity protocols class-1generation-2 UHF RFID protocol for communications at 8MHz–960 MHz, version 1.0.9. Sept (2005)

[6] Weis, S.A., Sarma, S.E., Rivest, R.L., Engels, D.W.: Security andprivacy aspects of low-cost radio frequency identification systems .In: Hutter D., Müller G., Stephan W., Ullmann M., (eds.) SPC,volume 2802 of Lecture Notes in Computer Science, pp. 201–212.Springer (2003)

[7] B. Gassend, D. E. Clarke, M. van Dijk, and S. Devadas. Silicon physical random functions. In Vijayalakshmi Atluri, editor, ACM Conference on Computer and Communications Security — CCS 2002, pages 148–160. ACM, November 18-22,2002.

[8] J. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas. A Technique to Build a Secret Key in Integrated Circuits for Identification and Authentication Applications. In VLSI Circuits Symposium, pages 176–179. IEEE Computer Society, June 17-19, 2004.

[9] T. Okamoto. Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In E. F. Brickell, editor, Advances in Cryptology —CRYPTO'92, volume 740 of LNCS, pages 31–53. Springer, 1992.

[10] A. Juels and S. A. Weis. Authenticating pervasive devices with human protocols. In V. Shoup, editor, Advances in Cryptology: Proceedings of CRYPTO 2005, volume 3621 of LNCS, pages 293–308. Springer-Verlag, 2005.

[11] A. Juels and S. A. Weis. Authenticating pervasive devices with human protocols. In V. Shoup, editor, Advances in Cryptology: Proceedings of CRYPTO 2005, volume 3621 of LNCS, pages 293–308. Springer-Verlag, 2005.

[12] A. Juels. Strengthening EPC Tags Against Cloning.In M. Jakobsson and R. Poovendran, editors, ACM Workshop on Wireless Security — WiSe 2005 ,pages 67–76. ACM Press, 2005.

[13] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In M. Joye and J. J. Quisquater, editors, Proceedings of 6th International Workshop on Cryptographic Hardware in Embedded Systems (CHES), volume 3156 of LNCS, pages 357–370. Springer Verlag, 2004.

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

[15] Kaufman, C., Perlman, R., and Speciner, M. Network Security: Private Communication in a Public World. 2nd ed. Upper Saddle River, N.J.: Prentice Hall PTR, 2002.

[16] Daemen, J., and Rijmen, V. AES Proposal: Rijndael. September 3, 1999. http://www.comms.scitech.sussex.ac.uk/fft/crypto/rijndael.pdf (accessed March, 15, 2010).

[17] Using Padding in Encryption. DI Management. January 3, 2010. http://www.di-mgt.com.au/cryptopad.html (accessed March, 15, 2010).

[18] M. Bellare, R. Canetti and H. Krawczyk, "Keying hash functions for message authentication," Advances in Cryptology – Crypto 96 Proceedings, Lecture Notes in Computer Science Vol. 1109, N. Koblitzed., Springer-Verlag, 1996.

[19] S. E. Sarma, S. A. Weis and D. W. Engels, "RFID systems, security and privacy implications," Technical report MIT-AUTOID-WH-014, AutoID center, MIT,2002

[20] S. Weis, S. Sarma, R. Rivest and D. Engels, "Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems," in 1st Intern. Conference on Security in Pervasive Computing (SPC), 2003.

.

[21] Oren, Y., Feldhofer, M.: WIPR—public-key identification on two
Grains of sand .In :Do minikus S.,(ed.)Workshop on RFID Security ,pp. 15–27 (2008)

[22] Rabin, M.O.: Digitalized signatures and public-key functions as in tractable as factorization. (1979)

[23] Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput.Syst. Sci. 28(2), 270–299 (1984)

[24]
Naccache,D.:Method,senderapparatusandreceiverapparatusfor modulo operation. US Patent 5,479,511, 26 Dec (1995)

[25] Shamir, A.: Memory efficient variants of public-key schemes for smart card applications. In: Advances in Cryptology-EUROCRYPT'94, pp. 445–449. Springer (1995)

[26] Shamir, A.: SQUASH-a new MAC with provable security properties for highly constrained devices such as RFID tags. In: Fast Software Encryption, pp. 144–157. Springer (2008)

[27] A Lightweight RFID Protocol to protect against Traceability and Cloning attacks, Tassos Dimitriou, tdim@ait.edu.gr

[28] Luby, M., Rackoff, C.: How to construct pseudorandom permutations from pseudorandom functions. SIAM J. Comput. 17(2),373–386 (1988)