RFID Security: Attacks, Countermeasures and Challenges

Mike Burmester and Breno de Medeiros
Computer Science Department
Florida State University
Tallahassee, FL 32306

{burmester, breno}@cs.fsu.edu

Abstract

Low-cost RFID tags are already being used for supply chain management and are a promising new technology that can be used to support the security of wireless ubiquitous applications. However current RFID technology is designed to optimize performance, with less attention paid to resilience and security. In this paper we analyze some of the most common types of attack on RFID tags: unauthorized disabling, unauthorized cloning, unauthorized tracking, and response replay.

We introduce security mechanisms appropriate to defeat these attacks, and show how a recently proposed RFID authentication protocol uses them to achieve security. Two implementations are considered, one using a shrinking generator, the other the AES block cipher. Both have small footprint and power-consumption characteristics, well within EPC constraints for tags with read-write capability (class 2). We conclude by discussing the need for a modular security approach with RFID technology that will support off-the-shelf applications, and the need for making RFID technology resistant to side-channel attacks.

I. Introduction

Radio-Frequency Identification (RFID) tags were initially developed as very small electronic hardware components having as their main function to broadcast a unique identifying number upon request. The simplest types of RFID tags are *passive* devices that not have an internal power source and are incapable of autonomous activity. They are powered by the reader's radio waves, with their antenna doubling as a source of inductive power.

While admittedly a new technology, the low-cost and high convenience value of RFID tags gives them the potential for massive deployment, for business automation applications and as smart, mass-market, embedded devices that support ubiquitous applications. However, current RFID protocols are designed to optimize performance, with lesser attention paid to resilience and security. Consequently, most RFID systems are inherently insecure. In this paper, we discuss four common types of RFID tag attacks that are particularly threatening.

Unauthorized tag disabling. These are *Denial-of-Service* (DoS) attacks in which an attacker causes RFID tags to assume a state from which they can no longer function properly. This results in the tags becoming either temporarily or permanently incapacitated.

Such attacks are often exacerbated by the mobile nature of the tags, allowing them to be manipulated at a distance by covert readers.

Tag disabling can be a serious threat to the integrity of automated inventory and shipping applications. Any RFID system vulnerable to such attacks could become a serious organizational weakness. Consider for instance the use of RFIDs to prevent shoplifting; in this case, the disabling activity might be performed covertly, avoiding detection through secondary mechanisms such as monitoring by cameras. If RFIDs are being used for automated inventory and/or shipping, it could again be a target of sabotage by competitors, paramilitary organizations (in the case of military shipments), militant activists, and/or terrorists.

Unauthorized tag cloning. These are *integrity* attacks in which an attacker succeeds in capturing a tag's identifying information. Again these attacks are exacerbated by the fact that the tags can be manipulated by rogue readers.

The ability to create clones of tags can be used as a means to overcome counterfeit protection (e.g., in passports and drug labels) and as a preparatory step in a (large-scale) theft scheme. Again, it exposes corporations to new vulnerabilities if RFIDs are used to automate verification steps to streamline security procedures.

Unauthorized tag tracking. These are *privacy* attacks in which the attacker can trace tags through rogue readers. We distinguish these attacks from "*Big Brother*" concerns that corporate entities managing the back-end server might leverage RFID capabilities to infringe on the privacy of consumers. A detailed analysis of consumer privacy concerns is given in [14], addressing policies, standards, and checks to protect consumer interests. In this paper we concentrate instead on the prospect of rogue readers, controlled by hackers or adversarial organizations, being used to monitor tags. This issue is more difficult to address, since hackers cannot be presumed to adhere to policies or standards, or to follow specified protocols.

Replay attacks. These are *integrity* attacks in which the attacker uses a tag's response to a rogue reader's challenge to impersonate the tag. The main concern here is in the context of RFIDs being used as contactless identification cards (in substitution of magnetic swipe cards) to provide access to secured areas and/or resources. In such applications, RFIDs can be more vulnerable than other mechanisms, again due to their ability to be read at a distance by covert readers.

RFID protocols must be lightweight, taking into account the severe constraints imposed on the available power (induced at the antenna), the extremely limited computational capabilities, the small memory size, and the characteristics of the IC design (e.g., number of gates available for security code). In particular, most RFID platforms can only implement highly optimized symmetric-key cryptography.

In this paper, we are mainly concerned with security issues at the protocol layer. We are not concerned with physical or link layer issues, such as the coupling design, the power-up and collision arbitration processes, or the air-RFID interface. For details on such issues, and more generally on standards for RFID systems, the reader is referred to the Electronic Protocol Code [10] and the ISO 18000 standard [17]. We do point out, however, that physical attacks such as jamming and collision attacks are major security concerns for RFID

applications. In Section 7 we shall discuss side-channel attacks and timing attacks—both types are physical attacks that target the protocol layer interface.

A highly desirable security feature for RFID technologies is *modularity*: RFID tags may be deployed in a variety of contexts with similar security characteristics. This widespread practice can nonetheless introduce vulnerabilities: For instance, protocols are often analyzed under the implicit assumption of operating in isolation, and therefore may fail in unexpected ways when used in combination with other protocols. Since RFID tags may be components of larger ubiquitous systems, it is preferable to pursue security analysis techniques that guarantee preservation of security when the protocols are executed in arbitrary composition with other (secure) protocols. This type of security is provided by formalizing and analyzing the security of protocols within the *universal composability* (UC) framework [5, 6, 7]. (An alternative formal models-type approach called *reactive systems* was proposed by Pfitzmann and Waidner [20, 21].) There are several RFID protocols that achieve this level of security by using lightweight cryptographic mechanisms [4, 23]. We shall discuss these in more detail in the following sections.

II. RFID Deployments

A typical deployment of an RFID system involves three types of legitimate entities, namely *tags*, *readers* and *back-end servers*. The tags are attached to, or embedded in, objects to be identified. They consist of a *transponder* and an *RF coupling element*. The coupling element has an antenna coil to capture RF power, clock pulses and data from the RFID reader. The readers typically contain a *transceiver*, a *control unit*, and a *coupling element*, to interrogate tags. They implement a radio interface to the tags and also a high level interface to a back-end server that processes captured data.

The back-servers are trusted entities that maintain a database containing the information needed to identify tags, including their identification numbers. Since the integrity of an RFID system is entirely dependent on the proper behavior of the server, it is assumed that the server is physically secure and not attackable. It is certainly legitimate to consider privacy mechanisms that reduce the trust on the back-end server—for instance, to mitigate the ability of the server to collect user-behavior information, or to make the server function auditable. In this paper, however, we shall not investigate such privacy attacks. These have been discussed extensively elsewhere. For an overview of measures and mechanisms that can be used to deal with privacy issues concerning back-end servers we refer the reader to [22]. Here we shall consider the servers to be entirely trusted.

III. Passive RFID tags

There are basically three types of passive RFID transponders.

Smart labels. These are class 1 basic memory devices that are typically Read-Only. They are capable of storing small amounts of data, sufficient for tag identification. Smart labels are low-cost replacements of barcodes and are used for inventory control. They function by backscattering the carrier signal from RFID readers. Smart labels are quite insecure: they are subject to both unauthorized cloning and unauthorized tracking, though in many cases are at least resistant to disabling attacks since they have a single

operational state.

Re-writable tags. These are class 1 tags with re-writable memory containing non-volatile EEPROM used to store user-and/or server-defined information. In a typical application [1], they store server certificates used to identify tags and are updated each time a tag is identified by an authorized reader. These tags can also store kill-keys, used to disable them. Despite this additional functionality, re-writable tags are still insecure: They are subject to unauthorized cloning, and unauthorized disabling, and in cases unauthorized tracking. Indeed a hacker (rogue reader) can record a tag's certificate and use it to impersonate the tag, track the tag (only until the next time the tag interacts with an honest reader outside the range of the attacker), and/or replace it with an invalid certificate, to disable the tag.

IC tags. These are class 2 smart tags with a CMOS integrated circuit, ROM, RAM, and non-volatile EEPROM. They use the integrated circuit to process a reader's challenge and generate an appropriate response. IC tags are the most structured tags and used with an appropriate RFID protocol they can defeat the attacks discussed in the Introduction. In the rest of this paper we show how this is done.

RFID tags are a challenging platform from an information assurance standpoint. Their extremely limited computational capabilities imply that traditional multi-party computation techniques for securing communication protocols are not feasible, and instead that lightweight approaches must be considered. Yet the robustness and security requirements of RFID applications can be quite significant. Ultimately, security solutions for RFID applications must take as rigorous a view of security as other types of applications. Accordingly, our threat model assumes malicious or Byzantine attacks.

Threat model. We adopt the Byzantine threat model. In this model all entities (tags, readers, back-end server) including the adversary (the attackers) have polynomially bounded resources. The adversary controls the delivery schedule of all communication channels, and may eavesdrop into, or modify, their contents. The adversary may also instantiate new communication channels and directly interact with honest parties. However, since the reader-server channels are assumed secure, and any assumptions about reader-server time synchronization are made explicit at protocol set-up, it is unnecessary to model adversarial interactions with reader-server channels.

IV. COUNTERMEASURES AND SECURITY GUIDELINES

4.1 Countermeasures

The disabling attack. In a disabling attack the attacker causes tags to assume a state from which they can no longer be identified by the back-end server. One way to prevent this is by having each tag share with the server a permanent (non-erasable) private identifying key k_{tag} (another way, which is however not suitable for low-cost tags, would be to use public-key cryptography). Then, when a tag is challenged by a reader, it will generate a response using this private key. Of course, it should be hard for an attacker to extract the private key

from the tag's response. For this purpose a cryptographic *one-way function* should be used.

This solution relies heavily on the assumption that the server is trusted and physically secured.

The cloning attack. To defeat cloning attacks it should not be possible for an attacker to access a tag's identifying data. Such data should be kept private. However for authentication, it should be possible for the back-end server to verify a tag's response. The response must therefore corroborate (but not reveal!) the tag's identifying data. This can be achieved by having the server share a private key k_{tag} with each tag, as in the previous case.

The tracking attack. Unauthorized tracking is based on tracing a tag responses to a particular tag. This can be prevented by making certain that the values of the responses appear to an attacker as random, uniformly distributed. In fact, since we are assuming that all entities of an RFID system have polynomially bounded resources, it is sufficient for these values to be *pseudo-random*.

Replay attacks. To deal with replay attacks the tag's response must be unique for every server challenge. To achieve this, the values of the server challenges and the tag responses must be unpredictable. One way to achieve this is to enforce that the answers be (cryptographically) pseudo-random.

4.2 Security guidelines

The countermeasures described above can be taken as guidelines for designing secure RFID applications. An RFID protocol requires at least two passes for (one-way) tag authentication: a challenge from the server and a response from the tag. If the tag initiates the protocol then we need at least three passes for secure tag authentication. For a minimalist approach one should aim for two passes.

The cost of generating the tag response must also be minimal, if we take into account the severe restrictions on resources for tags. However, this does not necessarily extend to the back-end server that typically does not have such constraints. In the next section we shall describe an RFID authentication protocol that adopts these guidelines.

V. O-TRAP: AN OPTIMISTIC TRIVIAL RFID AUTHENTICATION PROTOCOL

In this section we briefly describe O-TRAP, an RFID authentication protocol that was proposed in [4]. This protocol is *optimistic*, i.e., its overhead is minimal when the RFID system is not under attack. The protocol has two passes and is illustrated in Figure 1.

In this protocol we assume that all authorized RFID readers are linked to a back-end server by a *secure communication* channel (reliable and authenticated). Each tag stores two values: a private, long-term key k_{tag}, which it shares with the back-end server and a volatile identifying pseudonym r_{tag} which is updated each time the tag is challenged. The server has a database D in which it stores for each tag the pair of values (r_{tag}, k_{tag}) indexed by r_{tag}—see Figure 2.

Figure1: O-TRAP: Optimistic Trivial RFID entity Authentication Protocol.

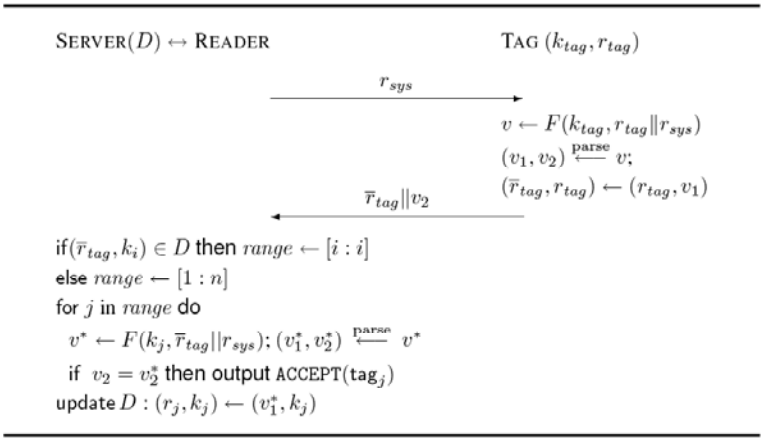

Figure 2: The database *D*.

r_{tag}	r_1	r_2	$\cdots$	r_n
k_{tag}	k_1	k_2	$\cdots$	k_n

At regular intervals, the server selects a random string r_{sys} that will be broadcast by the readers to all tags in their range.

Each tag, on activation by an RFID reader, computes two values v_1 and v_2, by applying the pseudo-random function F to (k_{tag}, r_{tag} || r_{sys}). The value v_1 is used to update the pseudo-random value r_{tag}; v_2 is used to authenticate the tag. When the adversary is passive, the server can retrieve the private key k_{tag} of the tag from its database D by using r_{tag}, and then verify the correctness of the tag's response and update the pseudo-random value r_{tag} corresponding to k_{tag} in the database D. In this case, the cost for both tag and server is just one application of the pseudo-random function F.

However, if the tag has most recently interacted with a malicious reader, then the stored values will be out-of-sync. In this case the server will have to exhaustively search through all private tag keys k_{tag} to find the correct value k_j and resynchronize, that is update in D the value corresponding to k_j, to the new value v_1.

Note that, in the presence of active attacks, the extra computational cost is borne out entirely by the server and not the tag. Also, note that the server challenge r_{sys} is the same for all tags in the range of the RFID readers during an interrogation period. During this period, the server must keep a list of tag replies, and reject replays. Authorized tags will not use the same reply. The server can manage the duration of the interrogation period to keep the replay list within reasonable length.

Optimizations for the adversarial case. O-TRAP is exceedingly efficient in the absence of active attacks, but reverts to a linear-search for the server when responses are tampered with by an active attacker. This can be mitigated by assigning multiple, replicated keys to tags,

with the effect of a (at most) linear increase in costs for the tags while the server search space decreases exponentially.

Security issues. It is clear that this protocol satisfies all the requirements set out in the guidelines for secure RFID applications (Section 4) if the function F is selected from a pseudo-random function family [15].

A formal proof of the security of O-TRAP in the UC framework is given in [4]. There are several other RFID protocols in [23] based on pseudo-random hash functions or pseudo-random bit generators that are provably secure in the UC framework.

One could argue that UC security is too much for low-cost RFID applications. The reason why we believe that this kind of security is essential for RFID applications, is that RFID protocols are not used in isolation, but concurrently, possibly involving other ubiquitous applications (e.g., sensors, motes, etc). O-TRAP shows that such level of security is achievable at a low cost.

VI. Implementation details

O-TRAP requires only the use of pseudo-random functions (PRFs). This results in a very flexible architecture since a variety of well-known and validated PRF constructions are established. Efficiency vs. security trade-offs in this architecture are easily achieved, as key-size and pseudo-randomness (estimated as the logarithmic length of the PRF cycle) can be chosen to the granularity of individual bits. Here we discuss two implementation strategies based on different PRF instantiations.

Using a well-known technique by Goldreich et. al. [16], it is possible to build a PRF that makes a call to a pseudo-random generator (PRG) per bit of input processed. In turn, a very efficient PRG implementation can be achieved using linear feedback shift registers, such as the self-shrinking generator [8]. This results in a small number of bit operations per input and output bits. Moreover, the entire footprint of the implementation can be fixed to require fewer than 2K gates to achieve 128-bit security [2], a range feasible for many RFID architectures (and within the EPC class2 constraints). A recently proposed implementation has achieved 128-bit security with only 1435 logic gates (within 517 clock cycles and 64B memory) [18].

Block ciphers can similarly be used to implement PRFs through a number of standard constructions [3]. When used only as PRFs, these constructions are in practice more efficient (in particular with regards to footprint) than security algorithms that require protocol parties to perform both encryption and decryption operations. Recently, highly optimized implementations of the Advanced Encryption Standard (AES) [9] block cipher algorithm have been achieved, and these are suitable for RFID architectures [13]. An RFID architecture using this implementation was proposed recently by [11], with footprint equal to 3,400 gates (in this implementation, gate complexity is based on 2-input NAND gates, called gate equivalents), and mean current consumption equal to 8μA, assuming a clock rate of 100kHz, and within 1032 clock cycles. Such implementations are more efficient than achievable by hash-based protocols, as demonstrated in [12].

VII. SIDE-CHANNEL ATTACKS AND TIMING ATTACKS

Side-channel attacks. A side-channel attack on an RFID systems exploits information leaked during its physical implementation, such as: timing information, power consumption, electromagnetic leaks, etc.

Side-channel attacks, and in particular power-consumption cryptanalysis, have been shown to be extremely effective, completely recovering cryptographic keys [19]. In order to achieve strong security in practice, research is needed into either making RFID hardware more resistant to such attacks, or developing obfuscating techniques for cryptographic computations.

An interesting theoretical question is whether physical security can be modeled within a UC framework— for example, by introducing information leakage channels and proving that such channels cannot give an advantage to adversaries, even in arbitrary composition and concurrency settings.

Timing attacks. In the case of O-TRAP, the tags and the back-end server take one computation step between sending and receiving authentication data. A secure implementation should reflect this semantic. In particular, the time taken for each pass must be constant. This can be done by inserting an artificial delay on the back-end server. This will not affect the throughput and workload of the server.

VIII. CONCLUSION

Strong security properties are achievable within simple security protocol designs that are suitable for implementation in RFID systems. In this paper, we described O-TRAP, a protocol for anonymous RFID identification that simultaneously achieves security against tracking, cloning, and disabling of tags, and that is not vulnerable to replay attacks. Recently, O-TRAP has been extended to provide forward-security [23].

ACKNOWLEDGMENT

The authors would like to thank Tri van Le for helpful discussions. He was a fundamental contributor to this research project and is a co-author in several of our related works.

REFERENCES

[1] G. Ateniese, J. Camenisch, and B. de Medeiros, *Untraceable RFID tags via insubvertible encryption*, Proc. ACM Conf. on Computer and Communication Security (ACM CCS 2005), ACM Press, 2005, pp. 92–101.

[2] Lejla Batina, Joseph Lano, Nele Mentens, Siddika Berna Ors, Bart Preneel, and Ingrid Verbauwhede, *Energy, performance, area versus security tradeoffs for stream ciphers*, The State of the Art of Stream Ciphers, Workshop Record, ECRYPT, 2004.

[3] Mihir Bellare, Anand Desai, Eron Jokipii, and Phillip Rogaway, *A concrete security treatment of symmetric encryption: Analysis of the DES modes of operation*, Proceedings of 38th Annual Symposium on Foundations of Computer Science (FOCS 97), IEEE Press, 1997, pp. pp. 394–403.

[4] M. Burmester, T. van Le, and B. de Medeiros, *Provably secure ubiquitous systems: Universally composable RFID authentication protocols*, Proceedings of the 2nd

IEEE/CreateNet International Conference on Security and Privacy in Communication Networks (SECURECOMM 2006), IEEE Press, 2006.

[5] R. Canetti, *Studies in secure multiparty computation and application*, Ph.D. thesis, Weizmann Institute of Science, Rehovot 76100, Israel, June 1995.

[6] —, *Security and composition of multi-party cryptographic protocols*, Journal of Cryptology 13:1 (2000), 143–202.

[7] —, *Universally composable security: A new paradigm for cryptographic protocols*, Proc. IEEE Symp. On Foundations of Computer Science (FOCS 2001), IEEE Press, 2001, pp. 136–145.

[8] D. Coppersmith, H. Krawczyk, and Y. Mansour, *The shrinking generator*, Proc. Advances in Cryptology (CRYPTO 1993), LNCS, Springer, 1994, pp. 22–39.

[9] Joan Daemen andVincent Rijmen, *The design of Rijndael*, Springer-Verlag, New York, Inc., Secaucus, NJ, USA, 2002.

[10] EPC Global, Inc., *EPC tag data standards, vs. 1.3*, http://www.epcglobalinc.org/standards/EPCglobal_Tag_Data_Standard_TDS_Version_1.3.pdf.

[11] Martin Feldhofer, Sandra Dominikus, and Johannes Wolkerstorfer,*Strong authentication for RFID systems using the AES algorithm*, Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems (CHES 2004), LNCS, vol. 3156, Springer, 2004, pp. 357–370.

[12] Martin Feldhofer and Christian Rechberger, *A case against currently used hash functions in RFID protocols*, Proceedings of On-The-Move Confederated International Workshops (OTM 2006), LNCS, vol. 4277, Springer, 2006.

[13] Martin Feldhofer, Johannes Wolkerstorfer, and Vincent Rijmen, *AES implementation on a grain of sand*, IEE Proceedings on Information Security 152 (2005), no. 1, 13–20.

[14] Christian Floerkemeier, Roland Schneider, and Marc Langheinrich, *Scanning with a purpose -supporting the fair information principles in RFID protocols*, UCS, 2004, pp. 214–231.

[15] Oded Goldreich, *The foundations of cryptography*, vol. 1, Cambridge University Press, 2001.

[16] Oded Goldreich, Shafi Goldwasser, and Silvio Micali, *How to construct random functions*. Journal of the ACM, 33 (1986), no. 4.

[17] ISO/IEC, *Standard #18000 RFID air interface standard*, http://www.hightechaid.com/standards/18000.htm.

[18] HangRok Lee and DoWon Hong, *The tag authentication scheme using self-shrinking generator on RFID system*, Transactions on Engineering, Computing, and Technology 18 (2006), 52–57.

[19] Yossi Oren and Adi Shamir, *Power analysis of RFID tags*, Appeared in the rump session of Advances in Cryptology (CRYPTO 2006), online at http://www.wisdom.weizmann.ac.il/~yossio/rfid/, Weizmann Institute, 2006.

[20] B. Pfitzmann and M. Waidner, *Composition and integrity preservation of secure reactive systems*, Proc. ACM Conf. on Computer and Communication Security (ACM CCS 2000), ACMPress, 2000, pp. 245–254.

[21] —, *A model for asynchronous reactive systems and its application to secure message transmission*, Proc. IEEE Symp. on Security and Privacy (S&P 2001), IEEE Press, 2001, pp. 184–200.

[22] S. E. Sharma, S. A. Wang, and D. W. Engels, *RFID systems and security and privacy implications*, Proc. of the Workshop on Cryptographic Hardware and Embedded Systems (CHES 20002), LNCS, vol. 2523, Springer, 2003, pp. 454–469.

[23] T. van Le, M. Burmester, and B. de Medeiros, *Universally composable and forward-secure RFID authentication and authenticated key exchange*, Proc. of the ACM Symp. on Information, Computer, and Communications Security (ASIACCS 2007), ACM Press, 2007.