2006

The Lazarus Effect: Resurrecting Killed RFID Tags

Christopher Bolan
Edith Cowan University

DOI: 10.4225/75/57b655b034765
Originally published in the Proceedings of 4th Australian Information Security Management Conference, Edith Cowan University, Perth, Western Australia, 5th December, 2006

The Lazarus Effect: Resurrecting Killed RFID Tags

Christopher Bolan
School of Computer and Information Science
Edith Cowan University
c.bolan@ecu.edu.au

Abstract

Several RFID Standards allow RFID Tags to be 'killed' using a specialised command code. 'Killed' RFID Tags should be irrevocably deactivated. In actuality, when a valid kill command is sent to a tag four sections of the tags management data are zeroed causing the tag to enter a 'fault state' and thus be ignored by RFID readers. Through the re-initialisation of these four sections to valid values it was discovered that a tag could be resurrected and function normally.

Keywords

Radio Frequency Identification, KILL, Reactivation

INTRODUCTION

RFID tags are typically silicon-based microchips. Functionality beyond simple identification upon request may be achieved including integrated sensors, read/write storage, encryption and access control (Weis, Sarma & Rivest, 2003; Bolan, 2006). The downside to such operations is the increased production cost of the RFID tag away from the ideal market penetration cost of $0.05 (US) - $0.10 (US), thus RFID security is often focused on reader security ignoring the obvious avenue of attack due to tag limitations (Choi, Lee & Lee, 2005).

Debate exists as to whether the adoption of some RFID Security measures are against the original vision of the technology. Knospe & Pohl (2004, p.46) argue that, as the primary purpose of RFID technology is as a cheap automated identification it is unreasonable to expect that standard security mechanisms be implemented, due to the complexity and constraints of the resource. Ranasinghe, Engels & Cole (2004) use this as a basis to propose that RFID Security be implemented at the data processing subsystem and thus leave RFID tags as merely identification. Though others (Engberg, Harning & Damsgaard-Jensen, 2004), argue that security is possible without effecting Tag cost or philosophical operations.

Irrespective of these arguments, no single security or encryption standard for tags or readers has been adopted and thus many systems remain insecure (Weis, 2003; Henrici & Müller, 2004). Noting such issues, Hennig, Ladkin & Sieker (2004, p.15-16) voice the following concerns:

- "*Worldwide unique IDs enable tracking*" – the adoption of unique EPC tags will allow anyone who carries at least one of these tags to be tracked worldwide.
- "*Unnoticed remote reading without line-of-sight*" – the very nature of RFID technology allows RFID tags to be read without line-of-sight or any overt suggestion that they are being engaged. Such features make unauthorised access more likely.
- "*Small hidden tags and readers*" – As tag sizes decrease the ease in which it becomes possible to install hidden tags, and readers increase.
- "*Tracking and profiling through sporadic surveillance*" – with a sufficient spread of strategically placed RFID readers it is possible to track and profile without the need for continual activation. Also, through the use of natural bottlenecks such as doorways it is further possible to ensure an individual passes within range of a Reader.

As a small step towards addressing some of these concerns Sarma, Weis & Engels (2002) proposed the creation of tags that allow a 'self-destruct' command which has subsequently been adopted on EPC Class One Generation 1 and 2 tags (EPCglobal, 2005). The feature requires the usage of a standard command and secret code that activates the logical destruction of a tag which upon receipt of the command ceases functioning. A 'self-destructed' tag would be irrevocably deactivated and thus never be re-activated (Juels, Rivest & Szydlo, 2003).

OPPONENTS AND DOCUMENTED WEAKNESSES

Fishkin, Roy & Jiang (2005) note that while such functionality is simple and effective there are two major weaknesses. Firstly, the command functions as an 'all or nothing' privacy mechanism, "*the tag responds to everyone until the kill switch is set, and then responds to no-one*" (*ibid*). The second weakness highlighted by Fishkin *et al.* (2005, p.3) is that "*the user has no way to know whether the tag has actually received the KILL command, let alone that the command was interpreted successfully*". Thus, while such a command would help allay some privacy concerns over active tags divulging data to illegitimate sources, it poses several risks.

The major risk of such functionality is that an attacker might utilise this function to permanently disable RFID tags. Such an attack would only require knowledge of the 'self-destruct' code and a suitable transceiver within range of the transponder that is to be deactivated. While such an attack was noted in the original design of the command in RFID systems, Sarma *et al*. (2002) suggested that a pervasive network of transceivers might be used to detect unauthorised 'self-destruct' commands. However, they fail to explain if such a system would then allow the blocking of the command or if the network would simply detect the command and let the transceivers be 'self-destructed'. An attack of this nature has been carried out by researchers at Edith Cowan University (Valli, Woodward, Bolan & Karvinen, 2006).

The inclusion of this functionality on tags is also challenged by Nathan *et al*. (2004) who query whether users of the technology will want to go to the effort to deactivate RFID tags. Also, there are questions as to the effect such functionality may have on consumers, who may want to keep RFID Tags active to use in household applications/systems (Spiekermann & Berthold, 2004; Engberg, Harning & Damsgaard-Jensen, 2004). In particular Ateniese, Camenisch & de Medeiros (2005), proffer the example of RFID enabled refrigerators that allow inventory access and automatic notification of food expiration, through the utilisation of existing tags.

TAG MEMORY

In order to fully understand the operation of killing and resurrecting an RFID Tag requires a basic understanding of RFID EPC Tag memory. The memory area of a EPC Tag is "*logically separated into four distinct banks, each of which may comprise zero or more memory words*" (EPCglobal, 2005, p.35). A diagrammatic representation of EPC tag compliant memory is given in figure 1.

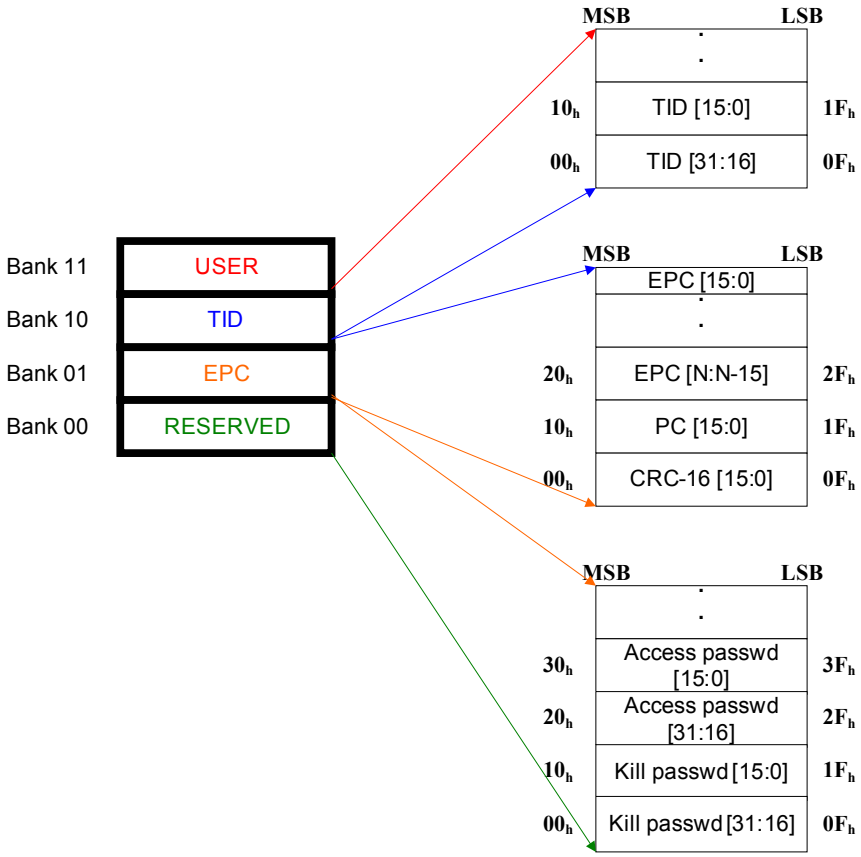

Figure 1. EPC Logical Memory Map (EPCglobal, 2005, p.35)

The first of the four memory banks, the reserved memory contains the kill (00_h - $1F_h$) and access (20_h - $3F_h$) passwords. The EPC memory block contains a CRC-16 (00_h - $0F_h$), Protocol-Control bits (10_h - $1F_h$) and an Electronic Product Code (20_h) identifying the object which the tag is (or will be) attached to. The Protocol Control is further divided into an EPC length field (10_h - 14_h), two 'reserved for future use' locations (15_h - 16_h) and a Numbering System Identifier (17_h - $1F_h$). The third memory area, 'Tag Identification' (TID) contains the 8-bit ISO/IEC 15963 allocation class identifier (00_h - 07_h). In addition, TID memory also contains sufficient identifying information to allow a Reader to uniquely identify any custom commands and/or optional features a given tag allows (08_h - $1F_h$) along with Vendor specific data ($1F_h$ above). The final logical area of tag memory is the least restricted 'User Memory' area. This area may be utilised in either a user defined or vendor specific fashion.

THE KILL COMMAND

The EPC Class 1 standard (EPCGlobal, 2005, p.58) specifies "*Interrogators and Tags shall implement the Kill command*" and further that the successful usage of the command will "*permanently disable a tag*". The actual 'KILL' instruction consists of eight bits (11000100) and is standard to all compliant tags, however the instruction is actually part of an overall command illustrated in figure 2.

	Command	Password	RFU	RN	CRC-16
# of bits	8	16	3	16	16
description	11000100	(½ kill password) ⊗ RN16	000_2	handle	

Figure 2. The EPC 'KILL' Command (EPCglobal, 2005, p.59)

The ‘KILL’ operation takes place as follows:

1. The Interrogator issues a Request Random Number (*Req_RN*) command
2. The Tag responds with a 16bit random number (RN) verified with a 16bit Cyclic Redundancy Check (CRC-16)
3. Using the acquired random number the Interrogator issues the KILL command using the Command (11000100), the most significant bits (bit range 31-16) EXOR the tag supplied random number, the Tags handle and a CRC-16
4. The Tag accepts the command and responds with its handle and a CRC-16
5. The Interrogator issues a second Request Random Number (*Req_RN*) command
6. The Tag supplies a new 16bit random number (RN) verified with a CRC-16
7. Using the acquired random number the Interrogator issues a second KILL command using the Command (11000100), the least significant bits (bit range 15-1) EXOR the tag supplied random number, the Tags handle and a CRC-16
8. If all steps were followed correctly the Tag responds with the KILL SUCCESS response (figure 3) after which it will “*render itself silent and shall not respond to an Interrogator thereafter*” (EPCglobal, 2005, p.58)

	Header	RN	CRC-16
# of bits	1	16	16
description	0	handle	

Figure 3. The KILL SUCCESS response (EPCglobal, 2005, p.59)

In investigation of RFID systems it was found that the successful running of a KILL command did not actually cease the functioning of an RFID tag. Once a KILL command had been successfully the Tag overwrites the Tag ID, CRC, Kill code and lock bits with 0 padded values. This is illustrated in figure 4.

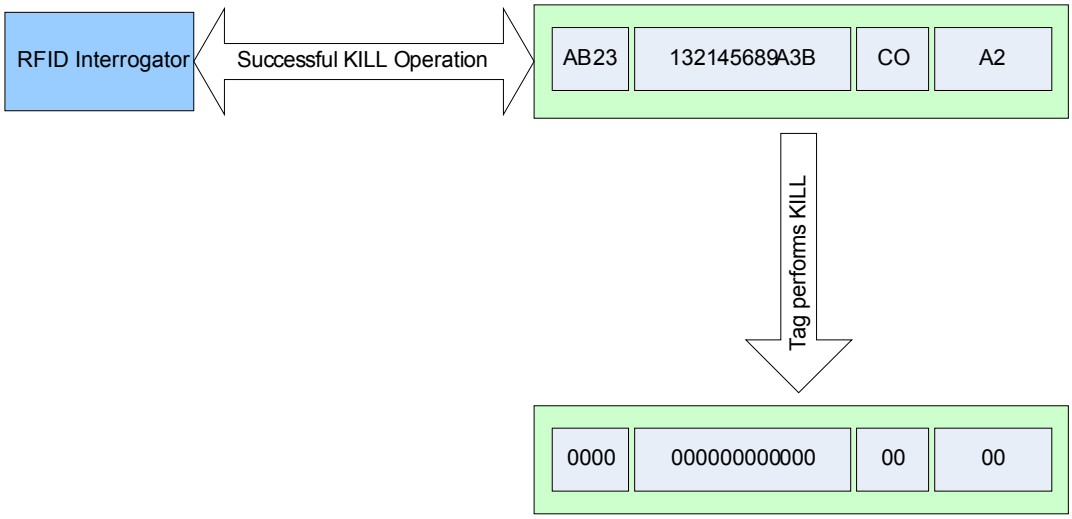

Figure 4. Successful KILL Operation

The next time a Tag is 'pinged' by an Interrogator the tag responds with its zeroed Tag ID and the zeroed CRC value. As the CRC value does not match the calculated value for the Tag ID the Interrogator effectively ignores the response and thus the tag is in essence 'Killed'. This finding contradicts the standards aforementioned claim that a Tag that has been 'Killed' will "*render itself silent and shall not respond to an Interrogator thereafter*" (EPCglobal, 2005, p.58).

REVIVING THE KILLED TAG

The method for tag resurrection is very simple, requiring the re-initialisation of the ID, CRC, Kill code and lock bits. First, the tag was read to ensure that the tag operated correctly and the initial tag values were recorded. Once the tag was verified the tag was then killed using the aforementioned 'Kill' command. The success of the 'Kill' operation was validated through the use of a standard RFID inventory application, which on repeated tests was unable to locate or interact with the killed tag. The killed tag was then overwritten with a new valid tag ID, CRC, Kill code and lock bits. The tag was then rechecked against the inventory application which was then able to read, verify and modify the resurrected tag. This process is illustrated in figure 5.

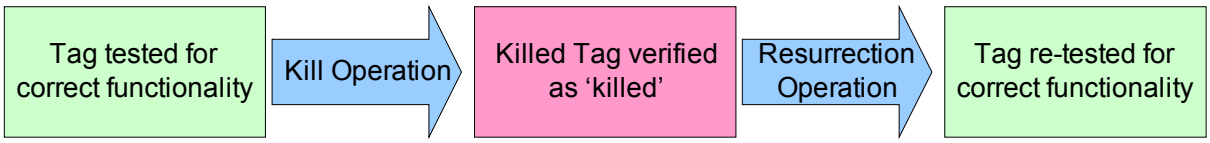

Figure 5. Experimentation Process

METHOD

An IBM laptop with a PCMCIA 11 WJC 6000 Gen1/2 reader and a Fractus EZConnect Embedded Antenna connected to the antenna 'A' socket was used as the reader during the tests. On the software side, a version of the WJC sample application provided as part of the MPR6000 RFID developers kit was modified to allow the individual operations (Read, Write, Verify, Kill) to be run manually as well as the logging of events to a Comma Separated Variable file 'Test.csv'. The augmented application is illustrated in figure 6 below.

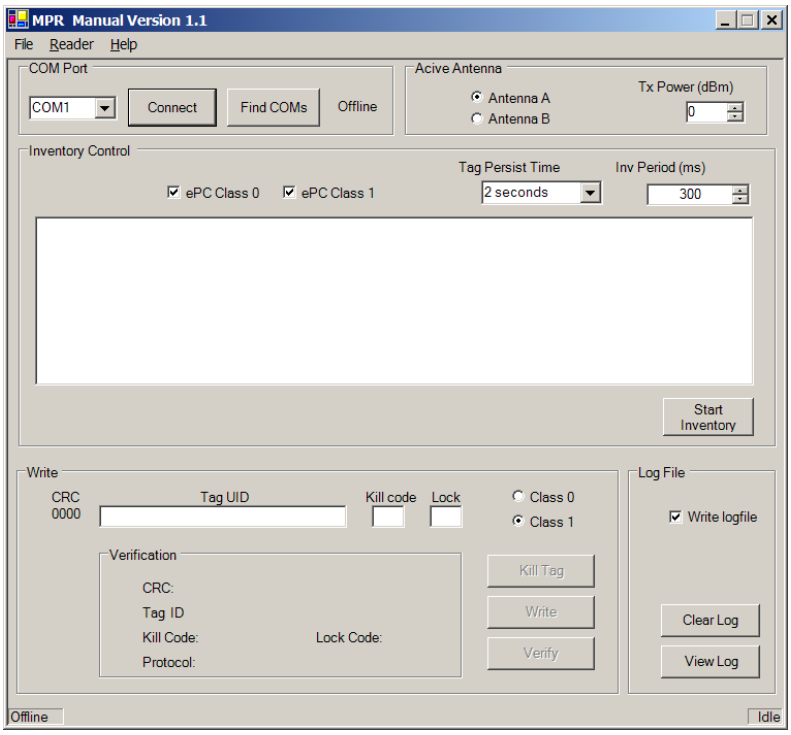

Figure 6. Modified MPR Application

Using the physical setup depicted in figure seven, a Tag was placed 10cm away from the EZConnect antenna and a Read command was issued to ensure the correct functioning of the Tag and to detail the values contained in the Tag ID, CRC, Kill code and lock bits.

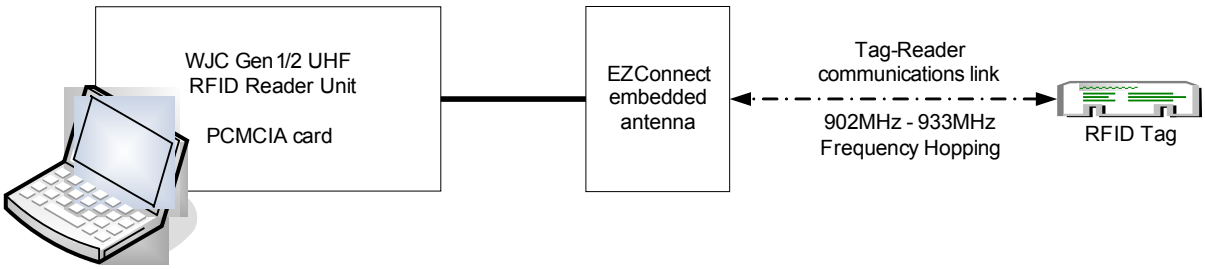

Figure 7. Experimental Setup

Once functionality had been established a 'Kill' command was issued using the modified software and the correct 'Kill' code. It should be noted that while separate research has been conducted by Edith Cowan University into a brute force 'Kill' attack dubbed 'Ken Kill'(Valli *et al.*, 2006), this paper is focussed on the functionality of the RFID Tag post 'Kill' operation and thus the 'Kill code' was used directly. As previously stated the Tag responds with KILL Success in figure 3 and then renders itself silent, to ensure the Tag has indeed been 'Killed' the application is then used to 'Verify' the Tag, which if the 'Kill' operation was successful the ID, CRC, Kill code and lock bits are returned as zeroed values as previously illustrated in figure 4.

After the 'Kill' is confirmed, the resurrection operation is attempted. Using the same application a new ID, CRC, Kill code and lock bits are transmitted to the 'Killed' Tag, in this experiment the values were reset to the new valid values. Upon completion of this operation a 'Verify' command is reissued and the refreshed Tag ID,

CRC, Kill code and lock bits are validated. Subsequently the Application is returned to a normal read or 'inventory' mode and the Tag reappears with its new Tag ID.

This process was carried out on a range of tags from different batches and suppliers. The resurrection of the tags was 100% successful in each test. While the author acknowledges that the experiment lacked strict environmental controls the strong initial results along with the support from the relevant documentation seem to indicate that the findings will be born out in further strictly controlled studies which are currently being undertaken at Edith Cowan University.

CONCLUSION

While the EPC standard emphasises (EPCglobal, 2005, p.40) "*Killed Tags shall remain in the killed state under all circumstances, and shall immediately enter killed upon subsequent power-ups. A kill operation is not reversible*", through a proof of concept and initial exploration it appears that despite such rhetoric, so called 'EPC compliant' RFID tags are able to be resurrected. It may be argued that the selection of RFID Tags used in the experiment were falsely claiming adherence to the EPC Standards as they fail to meet part a) of the conformance statement as outlined in the standard: "*A device shall not claim conformance with this specification unless the device complies with a) all clauses in this specification (except those marked as optional)*". However, such concerns are minimal as all selected Tags came with EPC compliance statements with the EPC certification logo and correctly followed all explicitly documented instructions in the standard, thus such an argument is tenuous.

The problem of Tag resurrection ultimately stems from the standard itself as nowhere beyond the previously discussed 'zeroing method' is the process by which a permanent 'kill' might actually occur documented. Despite this, even if unintentional the resurrection of a 'Killed' Tag may be seen by some as a beneficial 'feature'. For example a Tag on a supermarket item may be 'Killed' by the supermarket operator upon the sale of an item, allowing consumer privacy and possibly preventing further tracking of the purchase until the consumer returns home and resurrects the RFID Tag on the product to allow it to later interface with the consumers own RFID systems.

REFERENCES

Ateniese, G., Camenisch, J., & de Medeiros, B. (2005). Untraceable RFID Tags via Insubvertible Encryption. *Proceedings of the Conference on Computer and Communications Security - CCS'05*. Alexandria, Virginia, USA: ACM Press.

Bolan, C. (2006). Strategies for the Blocking of RFID Tags. *Proceedings of the Sixth International Network Conference*. Plymouth, UK

Choi, E. Y., Lee, S. M., & Lee, D. H. (2005). Efficient RFID Authentication protocol for Ubiquitous Computing Environment. In T. Enokido, L. Yan, B. Xiao, D. Kim, Y. Dai & L. Yang (Eds.), *International Workshop on Security in Ubiquitous Computing Systems - SECUBIQ2005* (Vol. 3823, pp. 945-954). Nagasaki, Japan: Springer-Verlag.

Engberg, S. J., Harning, M. B., & Damsgaard-Jensen, C. (2004). Zero-knowledge Device Authentication: Privacy & Security Enhanced RFID preserving Business Value and Consumer Convenience. *Proceedings of the Conference on Privacy, Security and Trust - PST*. New Brunswick, Canada

Engberg, S. J., Harning, M. B., & Damsgaard-Jensen, C. (2004). Zero-knowledge Device Authentication: Privacy & Security Enhanced RFID preserving Business Value and Consumer Convenience. *Proceedings of the Conference on Privacy, Security and Trust - PST*. New Brunswick, Canada

EPCglobal. (2005). *EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960MHz* (No. 1.0.9): EPCglobal.

Fishkin, K. P., Roy, S., & Jiang, B. (2005). Some Methods for Privacy in RFID Communication. In C. Castelluccia, H. Hartenstein, C. Paar & D. Westhoff (Eds.), *European Workshop on Security in Ad-hoc and Sensor Networks - ESAS 2004* (Vol. 3313, pp. 42-53). Heidelberg, Germany: Springer-Verlag.

Hennig, J. E., Ladkin, P. B., & Sieker, B. (2004). *Privacy Enhancing Technology Concepts for RFID Technology Scrutinised* (No. RVS-RR-04-02). Bielefeld, Germany: University of Bielefeld.

Henrici, D., & Müller, P. (2004). Tackling Security and Privacy Issues in Radio Frequency Identification Devices. In A. Ferscha & F. Mattern (Eds.), *Pervasive Computing* (Vol. 3001, pp. 219-224). Vienna, Austria: Springer-Verlag.

Juels, A., Rivest, R. L., & Szydlo, M. (2003). The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. *Proceedings of the 10th ACM conference on Computer and communications security* (pp.103 - 111). Washington D.C.

Knospe, H., & Pohl, H. (2004). RFID Security. *Information Security, 9*(4), 39-50.

Ranasinghe, D., Engels, D., & Cole, P. (2004). Security and Privacy: Modest Proposals for Low-Cost RFID Systems. In *Auto-ID Labs Research Workshop*. Zurich, Switzerland.

Sarma, S. E., Weis, S. A., & Engels, D. W. (2002). RFID Systems and Security and Privacy Implications. In *Workshop on Cryptographic Hardware and Embedded Systems* (Vol. 2523, pp. 454-470).

Spiekermann, S., & Berthold, O. (2004). Maintaining privacy in RFID enabled environments - Proposal for a disable-model. In *Workshop on Security and Privacy, Conference on Pervasive Computing*. Vienna, Austria.

Valli, C., Woodward, A., Bolan, C., & Karvinen, R. (2006). KenKill – A process to Kill Gen 1 Class 1 UHF RFID Tags. *Proceedings of the 4th Australian Information Security Management Conference*. Perth, Western Australia

Weis, S. (2003). *Security and Privacy in Radio-Frequency Identification Devices.* Unpublished Masters, Massachusetts Institute of Technology (MIT), Massachusetts, USA.

Weis, S. A., Sarma, S. E., Rivest, R. L., & Engels, D. W. (2003). Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In D. Hutter, G. Muller, W. Stephan & M. Ullmann (Eds.), *International Conference on Security in Pervasive Computing - SPC 2003* (Vol. 2802, pp. 454-469). Boppard, Germany: Springer-Verlag.

COPYRIGHT