VERSITA

Annales UMCS Informatica AI XI, 3 (2011) 9–25
DOI: 10.2478/v10065-011-0015-6

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

Security problems of systems of extremely weak devices

Marek Klonowski*

Institute of Mathematics and Computer Science, Wrocław University of Technology, ul. Wybrzeże Wyspiańskieo 27, 50-370 Wrocław, Poland

Abstract

In this paper we discuss some fundamental security issues of distributed systems of weak devices. We briefly describe two extreme kinds of such systems - the sensor network and theRadio Frequency IDentification (RFID) system from the point of view of security mechanisms designer. We describe some most important particularities and issues (including unsolved problems) that have to be taken into account in security design and analysis. Finally we present some fundamental concepts and paradigms of research on security of weak devices. In the paper we also give a brief survey of ultra–light HB/HB+ - family of encryption schemes and so-called predistribution protocols.

1. Introduction

Systems† of constrained devices are more and more common and their industrial/military importance is still growing. An obvious question is how to provide security for such systems. We bear in mind, in some sense, two extreme models - the first represented by a network of sensors, the other one is a system of RFID-tags (i.e. *Radio Frequency IDentification*). It is believed that ensuring

*E-mail address: Marek.Klonowski@pwr.wroc.pl

†Partially supported by Polish Ministry of Science and Higher Education, grant NN206 2573 35

an adequate level of security is a necessary condition of further development of systems of that type. On the other hand, one can observe that security issues are definitely neglected in many already implemented systems.

In this paper we discuss several main security problems of weak devices. We explain why constructing security mechanisms is so difficult for constrained devices. We also show that in many cases security mechanisms of these systems have to be completely different when compared to typical "high-end" systems.

We describe several typical techniques and tricks commonly used in such systems. In particular, we discuss *predistribution* protocols (typical of sensors) and family of HB protocols (designed for the RFID systems). We show that such security mechanisms are completely different from typical well-examined protocols like RSA. In particular, protocols and their analysis are based on different (compared with typical cryptography) metamathematical objects and tools.

1.1. Organization of this paper

Section 2 is devoted to description of systems of weak devices with particular focus on security issues. In Sections 3 and 4 we describe basic security problems, attacks as well as countermeasures. In Section 5 we discuss predistribution schemes for wireless sensor networks. In Section 6 we describe family of HB authentication protocols. Non-algorithmic, yet very important issues are discussed in Section 7. We conclude in 8.

2. Particularities of systems of small devices

In this section we briefly outline two, in some sense extreme, kinds of systems of weak devices - the sensor network and the RFID system. We point out to their peculiarities and applications. Then we try to explain why providing security to systems of weak devices is so demanding in general.

Sensor network. - by this term we mean a system of small devices distributed usually over a very large area in order to measure some environmental features (e.g temperature, humidity). Such system is usually very long lasting - in some cases it can be used for many years without any supervision in an open environment. Devices (sensors) have moderate memory (enough to keep several cryptographic keys) and computational power that allows to find values of one-way (in practice) hash function (e.g. SHA-256). In realistic time it is not feasible, however, to perform asymmetric cryptography. Sensors usually have a battery that in practice cannot be replaced. For these reasons algorithms need possibly to reduce communication - especially broadcasting. In some scenarios all sensors can communicate directly - in such a case we call the network *single*

hop. Otherwise, in a *multi-hop* network some nodes have to communicate via intermediate nodes.

Sensor networks are used not only for monitoring the environment but also for military purposes (e.g detecting movement of troops in a battlefield). Extended description of sensor networks and their applications can be found for example in [**1**].

RFID. - Radio Frequency IDentification are the systems composed of ultra weak devices called *tags* (or *transponders*) and so-called *readers* (or *transceivers*). A tag can be seen as a piece of memory that can be read from a short or moderate distance. They have minimal (if any) computational power that allows to perform basic operations (addition, xoring of small numbers). In extreme cases they have no computational power at all - they just respond with a static identifier on each reader's query. Readers have abilities (computations, memory, energy) comparable with the regular PC. In the system the reader is connected with a back end data-base that collects all data about tags. Thus the reader communicating with the tag can recognize it and then get much more information about it from the data-base.

Most of the tags, (e.g., so-called *passive tags*) do not have any inner source of energy. The energy necessary for replying the reader's query is supplied by RF of the reader. It is commonly assumed that the "middle-class" tag of reasonable size and cost e.g. the EPC class 1 should not have more than $2k$ logical gates. Note that it is not feasible to implement regular symmetric encryption protocol using such resources, since regular logical circuit realizing symmetric encryption protocol needs more than $20k$ logical gates. Even extremely elaborated implementations of AES need more than $5k$ logical gates ([**2**]). Extended description of the RFID system with particular attention to application and security threads can be found in [**3**].

The RFID systems were initially designed as successors of Universal Product Code (UPC) (the barcode found on most consumer products) in logistic chains. Thanks to the RFID-tags one simultaneously lists all products in the box without first unpacking all items. Recently, RFID has grasped much wider applications in sorting or detecting a variety of objects including goods, animals or even people. In particular, they are used for logical/physical access control, speed control, payment systems or even localizing devices for securing tickets or high value chips in casinos.

Noondays there are many thousand of millions of already produced RFID-tags [**3**]. Most of the tags are standardized as the Electronic Product Codes (EPC)-tags ([**4**]). This is an open and flexible standard that assigns each item a globally unique label.

There were some very spectacular implementations of RFID technology - for example in the US Department of Defense, Procter and Gamble and Wal-Mart chain, where the technology has made some logistic processes three times faster ([**5**])

In fact, it is not feasible to precisely distinguish "regular systems" from "systems of small devices" - there are many systems of intermediate capabilities that should be placed between the RFID-tags and the sensors. The fact remains, however, that there are some systems that need special algorithmic approach due to their specific features and strict constraints. Such systems very often process important data that have to be protected. Nevertheless, there are some peculiarities that make system of weak devices hard to protect in a proper way.

- Devices have constrained computational power and memory. Thus it is not possible to use advanced cryptographic methods. Similarly, due to the constrained memory, it is not feasible to use methods with limited computation but high memory requirements (e.g. one-time signatures).
- Energetic constraints cause limitations in communication (broadcasting). Thus in some models limited communication is one of the most important evaluation metrics of the algorithm ([**6**]). Energy resources are also very important in defining the adversary. Indeed, in some cases it is not possible to provide a high level of security if the adversary has unlimited energy. On the other hand, it is possible if we assume some constraints on the number of adversarial broadcasts [**7**].
- Typically, case systems of constrained devices work on a very vast (even unpredictable) area that cannot be supervised. Thus constructing mechanisms needs to take into account that the adversary can have physical access to parts of the system. For that reason, it is a realistic assumption that some devices can be seized and corrupted by the adversary. On the other hand, the legitimate user cannot simply replace damaged device or change the code executed by them.
- Devices communicate using a radio channel - thus one needs to take into account that sent messages may be eavesdropped and the connection is not reliable (due to e.g interferences, noise).

Such systems are distributed and thus inherit all problems typical of security issues of distributed systems including possible delays and synchronization problems and in some models difficulties induced by multiple access channel or limited bandwidth , etc. Additionally, sensor networks work often in a so-called *ad hoc* mode - for example they are spread from the plane. Thus we have no a priori knowledge of network configuration after deployment. Devices are

placed at random and some of them may be even destroyed. It is clear that such a network needs to execute some self-organizing preprocedure.

As we can see, designing security mechanisms for low-end devices is a very demanding task. On the other hand, there are only few peculiarities that make design of systems easier.

- size of processed data is usually significantly smaller when compared with a regular system - thus usually one can use very particular mechanisms designed only for short or moderate length messages.
- since in most of scenarios the system works on a large area we can safely assume that the adversary can control/eavesdrop only some parts of the system at the same time. ([**8**])

3. Security aspects of systems of weak devices

As we can see, constructing algorithms for the system described in the previous sections is much more difficult than for the regular ones due to very strict constraints and assumptions implied by practical demands. One may be tempted to neglect importance of security methods for small devices since smaller devices are more and more powerful. One can imagine that finally the RFID-tag would be have power of a regular PC.

On the other hand, there is still the need of creating smaller tags - note for example μ - tag constructed by HITACHI [**9**]. Even though this device is smaller than half square millimeter, there are still plans to make its second, much smaller generation. Additionally, there is price pressure to produce cheaper and cheaper tags. Indeed - industry needs to label cheaper and cheaper products. It is hard to imagine that one could accept a tag more expensive than the object bearing it.

3.1. Aims and possible threads

Generally speaking security aims of the system based on small devices are the same as in the regular "high-end" systems. At first, some of typical security properties will be discussed. Most of the security research devoted to weak devices discusses:

- confidentiality - only legitimate parties should have access to the data;
- integrity - data cannot be changed in an illegal way;
- availability - the legitimate user should always be able to have access to data;
- privacy of users - understood in many ways. In most cases this property boils down to preventing any party from getting any additional knowledge.

In literature many attacks against the described systems have been discussed so far. Some interesting examples are:

- Sybil attack - adversary tries to gain influence on the system more significantly than proportional to the resources under its control. For example, a single physical device can pretend several identities have greater influence on the whole system ([**7**]).
- DoS - the aim of the adversary is to make executing the protocol/ access to some data impossible.
- Capturing - the adversary may take control over some devices and change their behaviour (e.g using a modified code)
- Cloning - the adversary can make copies of devices.

Privacy threads. Attacks against privacy are, in principle, discussed only in the context of RFID-systems. In the basic scenario the problem with privacy is as follows - the bearer of an object with the tag can be easily traced. Moreover, such remote tracing can be in practice unnoticeable. Combining information from the tag with other data (e.g time, patterns of behaviour) can reveal some other information about individuals. There are many lines of protecting from this threat. One of possible methods is " kill command" – e.g. the tag is permanently switched off (e.g. after an item is sold). However, this strongly limits expected functionality of the system. Another approach is putting the tag in a Faraday cage or using special devices that generate noise and jam communication. Thus, as long as the jammer is switched on, no communication with the tag is possible. Both solutions are not feasible in most applications. In [**10**] the lightweight method called *pseudonym throttling* has been suggested wherein passwords are cyclically changed. The Reader can communicate with the tag only after sending a password. The adversary has to eavesdrop many times to carry out a successful attack. The survey of similar methods is given in [**3**].

It should be stressed, however, that none of these methods may ensure perfect privacy. Indeed, note that even encrypted communication if detected by the adversary can reveal some information.

4. Light–weight methods

Security mechanisms for constrained devices have gained significant attention for many years. There is a long list of protocols that try to provide security protection methods with computational/memory/communicational constraints. This line of research is sometimes called *light–weight cryptography*. Such protocols were motivated not only by constraints of devices but also the need of having extremely efficient protocols or even to by-pass some legal restrictions

([**11**]) In most cases the presented methods are based on computing values of a one-way hash function using constant-size memory.

Having solely the ability of computing values of one-way hash function one can implement quite a rich collection of protocols, including digital signatures e.g. Lamport signatures ([**12**]) or more efficient in terms of memory Winterlitz signatures. In Section 5 we recall protocols for key-establishment and key management. Of course, one can also implement various challenge-response protocols. Some elaborated constructions are surveyed in [**1**]. There are, however, some theoretical limitations of functionality that can be accomplished using only one-way hash functions (e.g. [**13**]).

The protocols presented above, suitable for sensors, are out of reach of typical RFID-tags. For those devices some special protocols are designed. An example of such ultra-lightweight approach is given in Section 6.

Solutions for low-end RFID-tags. One needs to take into account that most tags have **no** computational power at all. Of course, they cannot implement any even ultra light-weight cryptographic methods. Fortunately, there are some methods that allow to implement security mechanisms even for such devices. Of course, such security is very limited.

The methods described below do not require any computations on the tag's side. They can be also used together with other mechanisms to support the overall level of security. One approach is to use a very slow-charging capacitor in the tag. Such tag needs significant amount of time to collect energy for answering the reader's query. Thanks to it, in order to read the tag the reader needs longer time of communication. This, to some extent, protects the users from unwanted reading of tags they possess - especially if the tags are in move.

Even more important is the fact, that a number of queries the tag can answer is strictly limited. In realistic scenarios this feature does not restrict usability. On the other hand, it protects from some attacks wherein the adversary needs to collect a sufficient, usually large number, of tag answers [**14, 15**].

Another non-algorithmic way of securing the tag was presented in [**16**]. In this paper the authors propose to remove part of the tag antenna to reduce the distance of reading. The idea is very realistic, however, only in some scenarios. For example in a shop, after the an item is sold, a shortened antenna protects the buyer's privacy. On the other hand, if necessary, the tag can be read (e.g. customer's claim) but only from a very short range.

Some physical methods are very effective but their application is limited to a very particular case. For example in some countries (including Poland and the UK) there are RFID-tags in passports. The cover of the passport has embedded

a Faraday cage. If the covers are open the tag can be read. Otherwise, covers protect from unwanted reading.

Paper [**17**] describes a protocol that requires that memory of the tag can be overwritten. Idea of the protocol is as follows – readers in each communication change slightly the state of the memory. Legitimate readers can easily trace tags, however if the passive adversary (eavesdropper) skips some changes, it cannot recognize the traced tag anymore. Security of this simple protocol is reduced to the analysis of a random walk on a hypercube.

5. Predistribution schemes

Devices like sensors can efficiently use symmetric ciphers like AES. How to encrypt messages in a sensor network using only symmetric ciphers ? The problem is with the proper key establishment. One of possible approaches is to choose before deployment one single secret key shared by all devices. As long as the adversary is passive, confidentiality of exchanged messages is guaranteed. The adversary can, however, decrypt all messages in the network seizing even a single device. Due to vulnerability of nodes to physical capture such strategy is unacceptable. Another extreme (and naive) approach it to give each sensor a different key. In such scenario, devices cannot decrypt exchanged messages without exchanging keys. This, however, would demand asymmetric methods. One of the most exploited lines of research on security for weak devices in the context of sensor networks is *predistibution*, that can be seen as a trade-off between those two strategies. The idea is as follows. At the beginning a pool of secret keys K of cardinality k is chosen. Before the devices are placed in the target environment, each of them gets a subset of K of cardinality n. If $n = \Theta(\sqrt{k})$ is chosen carefully, then each pair of devices has at least one common key w.h.p (due to birthday paradox) and can exchange messages. On the other hand, an adversary capturing a single sensor is not able to decrypt all messages exchanged in the network. That is - the attack is local. This idea was introduced in the seminal paper [**18**].

The first improvement of this protocol was proposed in [**19**], wherein the authors suggested to force sensors to use $q > 1$ keys to initiate communications. Its a results, two sensors should have at least q common keys to communicate. Moreover, each message is encrypted using q common keys. Thus each pair of sensors uses one of $\binom{k}{q}$ instead of one of k combination keys from K. The adversary capturing a single sensor is very unlikely to be able to decrypt communication of any other pair. On the other hand, such approach has also some drawbacks - to have at least constant probability (with respect to k) that two

sensors have q common keys, we need $n = \Omega(k^{\frac{q-1}{q}})$. Even for moderate q sensors need much more memory. What is more important, the adversary seizing any sensor collects more nodes. It turns out that this approach is better only if we assume that the adversary can collect a small number of sensors.

Small, yet very efficient improvement was proposed [**20**]. In this paper the authors suggest treating a set K as a projective space. Instead of random subsets of K, each sensor gets exactly one line of a projective space. Thus every two sensors have exactly one common key.

Another approach, called *Multipath Key Reinforcement*,presented in [**19**] suggests routing messages through several indeterminate nodes. To decrypt a message, the adversary needs much more keys.

Key-evolution protocols. Another interesting notion strongly connected with predistribution schemes is *key evolution protocols*. This mechanism is simple, yet very effective. The idea behind it is as follows - sensors use a fixed key just for a short period. In the next period the key is changed in a pseudo-random manner. The adversary that captures some keys in period t is not able to decrypt messages in period $t + 1$ (*forward security*) or $t - 1$ (*backward security*). In paper [**6**] a key evolution protocol with forward security has been proposed. Protocol from [**21**] offers both forward and backward security. It is interesting that the key evolution protocols can be constructed without asymmetric cryptography using only one-way hash function.

6. HB/HB+ -type Protocols

In this section we describe one of the most important types of ultra-lightweight authentication scheme, called HB. A very simple construction and a clear mathematical model made that the basic scheme and its numerous extensions gained significant attention in literature. The HB protocol was introduced in the context of RFID-systems in [**22**], however, it is based on *human-to-computer authentication protocol* designed by Hopper and Blum ([**23**]). In principle, HB is an authentication protocol - i.e., the tag would like to authenticate itself to the reader.

6.1. HB-protocol description

Let $x = x_1, x_2, \ldots x_n$, $y = y_1, y_2, \ldots y_n$ be two bit-stings of the length n. We define a dot product $< x, y >= \sum x_i y_i \mod 2$. By $x \oplus y$ we denote a bit-wise xor of strings x, y. Let $x \in_R A$ denote an element x chosen at random from the set A.

In the HB protocol, a secret key is a n-bit string x shared between the reader and the tag. Additionally, the protocol depends on two parameters N - i.e., a number of rounds and a parameter $\varepsilon < 1/2$.

HB protocol
Public parameters: n
Secret key: $x \in_R \{0,1\}^n$
Parameter: $L = 0$
Following procedure is repeated N times:

Reader		**Tag**
Chooses $a \in_R \{0,1\}^n$	$\xrightarrow{a}$	
		Sets $v := 1$ with prob. ε, else $v := 0$
		Computes $r :=< x, a > \oplus v$
	$\xleftarrow{r}$	

If $r =< x, r >$ than $L := L + 1$;
The tag is accepted iff $L \geq (1 - \varepsilon)N$

6.1.1. HB-security of the protocol

The basic considered security model is as follows – the tag and the reader share a key. The adversary has to prove its identity to the reader. The adversary eavesdrops communication between the tag and the reader and its aim is to clone the tag or at least to mimic the legitimate tag. The adversary cannot seize the tag or communicate with the tag or the reader.

Under such conditions the security of the HB schemes is provable. More precisely *learning parity with noise* problem (LPN) is shown to be reducible to finding the secret key of HB. LPN itself was proved to be NP-hard [**24**]. Inapproximability of this problem (with factor two) has been shown in [**25**].

Despite the reduction, security analysis is not fully convincing - this reduction does not tell about security of a typical instance of the problem. Moreover, in recent years there have been several attacks against the LPN problem. Most of them (e.g. *LF2* from [**26**] or [**27**]) are tune-ups of the BKW algorithm (Blum, Kalai, Wasserman 2003) [**28**] that has **subexponential** runtime $O(2^{k \log k})$. The fact remains, however, that the reduction is a relatively strong evidence for using the HB protocol.

One can easily observe that assumptions are important. If the adversary could adaptively query the tag, it would retrieve the key bit by bit using $\Omega((1-2\varepsilon)^{-2})$ queries and simple Gaussian elimination (w.h.p.).

It is believed that the HB protocols protects also privacy, nevertheless this property has not formally been proven up to now.

6.2. HB+

HB+ is an augmented version of the HB protocol and was proposed already in [**22**]. As mentioned in the previous subsection, HB protects only against passive adversary limited to eavesdropping communication between a legitimate tag and a legitimate reader. The new version of the protocol was designed in order to protect the communication against an active adversary.

HB protocol
Public parameters: n
Secret key: $x \in_R \{0,1\}^n, y \in_R \{0,1\}^m$
Parameter: $L = 0$
Following procedure is repeated N times:

Reader		**Tag**
		$b \in_R \{0,1\}^m$
	$\xleftarrow{b}$	
Chooses $a \in_R \{0,1\}^n$		
	$\xrightarrow{a}$	
		Sets $v := 1$ with prob. ε, else $v := 0$
		Computes $r :=< x, a > \oplus < y, b > v$
	$\xleftarrow{r}$	
If $r =< x, r > \oplus < y, b >$ than $L := L + 1$;		

The tag is accepted iff $L \geq (1 - \varepsilon)N$

The exemplary parameters considered in practice are as follows $\varepsilon = 1/4$, $N = 1164$. The lengths of x and y are 80 and 512 respectively.

As one can see the main difference is that HB+ protocol is a commitment-challenge-response protocol. The tag sends a vector b as an additional blinding factor . It was proved in [**22**] that HB+ is secure against active adversary in the so-called *detection-based model* (LPN reduction). The extended security analysis of this scheme was shown in [**29, 30**] also for some modifications of HB+. However, the assumed model does not take into account that the adversary can communicate with the reader. It turns out that the adversary capable of communicating with the reader can mount a man-in-the middle attack in a very efficient way as shown in [**31**]. The attack can retrieve the key in a linear number of queries. Some other attacks against this protocol have also been shown in [**32**] and [**33**].

6.3. Other schemes from the HB family

Next, a natural step in research was to construct a protocol immune against Man-in-the-middle attack. There have been several published protocols so far. Below we list the most notable of them.

HB#: was introduced in [**34**]. The proposed solution was based on the Toeplitz matrices in order to make the scheme even more efficient in

terms of computation. Some security shortcomings of this scheme were shown in [**35**] .

HB-PUF: The protocol introduced in [**36**] is based on Physically Unclonable Function (PUF). The PUF mechanism is based on small, " unpredictable" in practice variances typical of each integrated circuit. It is assumed that two circuits have different responses as PUF even if they are logically the same. It is hard to judge how practical the solution is.

Trusted-HB: This protocol was introduced in [**37**]. It combines the basic HB+ with lightweight signing mechanisms from [**38**]. The protocol was, however, efficiently attacked in [**39**] .

HB++: The protocol was shown in [**40**]. It runs regular HB+ protocols with different secrets and correlated challenges. The protocol was attacked in [**41**].

HB-MP: Has been presented in [**42**] as a more immune and more efficient version of HB+ protocol.

6.4. Similar methods

There are some protocols offering similar functionality and having similar requirements. One of them is CKK ([**14**]). This protocol offers much weaker security (in terms of a number of eavesdropped queries), however the security analysis is very precise. Extensions of the basic CKK scheme were attacked in [**43**] and [**15**].

7. Theory vs. Practice

Except for algorithmic issues discussed in the previous sections there are also many other problems in implementing security methods in the system of weak devices that need to be handled.

7.1. Source of randomness

Security algorithms strongly depend on quality of randomness. In many cases if the adversary has access to a source of randomness generated locally in devices, then breaking the system is a trivial task. In some cases if the source of data is even slightly biased, then the adversary can carry out very efficient attacks. How to generate random bits in weak devices? Of course, a sensor cannot use classic methods like those based on HDD movement. In the case of devices of moderate capacities a reasonable thing to do is to use a one-way hash function with random seeds r_0, s. Then the i-th random sequence can be computed as $r_i = H(r_{i-1}, s)$. The quality of randomness can be improved using

deterministic functions, the so-called *extractors* if several independent sources of random bits (possibly of low quality) are available.

Such generation of random bits is out of reach even for the advanced RFID tags. In research papers published recently we can find several approaches. The most typical is to use some external noise (e.g. thermal noise) and amplify it using a simple electronic circuit. Somehow similar idea is based on an accelerometer integrated with the tag. Both methods can be somehow problematic in very static systems.

Substantially different and very promising is a method called FRENS proposed in [**44**]. The authors propose to use the initial state of SRAM (Static Random-Access Memory) that tends to be very unstable and, to some extent, "unpredictable".

To the best of our knowledge there is no provably secure method of random bit generation implemented in the RFID-tags.

7.2. Non–technical problems

Another issue is social acceptance of ubiquitous technology. In particular, RFID-technology raises a number of concerns regarding privacy. It should be noted that in many cases even issuers of tags or manufacturers do not know what is the effective range of communication with the tag. Moreover, customers may not know if they have any tag or if it is still functional. This is not only a problem of personal location privacy but also combining data from the tag with other data that can together reveal a lot of sensitive information about individuals. Due to possible DoS attack or risk of industrial espionage RFID technology is also found insecure in some corporations. It is not clear if the social acceptance of the technology will be significantly higher when security mechanisms are better.

Fig. 1. Logo of the anti-RFID campaign by German privacy group FoeBuD.(From Wikipedia).

Another problem is with standardization. One can see that there are several organizations dealing with the RFID-tags. The most prominent, except EPCGlobal are the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). Thus there is no single specification - both in data representation and hardware requirements.

8. Conclusions

We presented some problems and methods related to security in the distributed systems of weak devices. Most of well-known methods designed for standard devices like PCs cannot be implemented for constrained devices like the RFID-tags or even sensors. It should be stressed that this is not only a problem of constrained resources but also other factors like different models of the realistic adversary.

When comparing with typical systems, those of weak devices have completely different nature. In particular, one cannot expect the level of security as high as in typical "high-end" systems. One should rather demand *possibly* high security using in advance strictly constrained resources. Another interesting observation is that protocols designed for weak devices use usually substantially different mathematical objects as underlying structures - both in construction and the analysis.

We believe that further development of such systems is strongly dependent on providing adequate (in terms of cost, robustness as well as usability) security mechanisms. The proposed methods definitely do not fulfill all expected requirements and many important questions are left unanswered (in particular, in strongly distributed systems like multihop sensor network and highly dynamic networks). Last but not least, providing a fair level of security to the systems of weak devices requires not only proper, very fancy algorithms but also taking into account other factors like physical and organizational (legal) features.

References

[1] Błaśkiewicz P. and Kutyłowski M. *Security and Trust in Sensor Networksin Theoretical Aspects of Distributed Computing in Sensor Networks Jose D.P. Rolim, Sotiris Nikoletseas (Eds.)*. Springer Verlag, 2011.

[2] Feldhofer M., Dominikus S., and Wolkerstorfer J. Strong authentication for rfid systems using the aes algorithm. In Marc Joye and Jean-Jacques Quisquater, editors, *CHES*, volume 3156 of *Lecture Notes in Computer Science*, pages 357–370. Springer, 2004.

[3] Juels A. Rfid security and privacy: a research survey. *IEEE Journal on Selected Areas in Communications*, 24(2):381–394, 2006.

[4] `http://www.gs1.org/epcglobal/standards`.

[5] `http://www.infosec.gov.hk/english/technical/files/rfid.pdf`. preprint, 2008.

[6] Ren M., Das T. K., and Zhou J. Diverging keys in wireless sensor networks. In Sokratis K. Katsikas, Javier Lopez, Michael Backes, Stefanos Gritzalis, and Bart Preneel, editors, *ISC*, volume 4176 of *Lecture Notes in Computer Science*, pages 257–269. Springer, 2006.

[7] Golebiewski Z., Klonowski M., Koza M., and Kutylowski M. Towards fair leader election in wireless networks. In Pedro M. Ruiz and Jose Joaquin Garcia-Luna-Aceves, editors, *ADHOC-NOW*, volume 5793 of *Lecture Notes in Computer Science*, pages 166–179. Springer, 2009.

[8] Berman R., Fiat A., and Ta-Shma A. Provable unlinkability against traffic analysis. In Ari Juels, editor, *Financial Cryptography*, volume 3110 of *Lecture Notes in Computer Science*, pages 266–280. Springer, 2004.

[9] `http://www.hitachi.pl/ifg/Products/muchip.html`.

[10] Juels A. Minimalist cryptography for low-cost rfid tags. In Carlo Blundo and Stelvio Cimato, editors, *SCN*, volume 3352 of *Lecture Notes in Computer Science*, pages 149–164. Springer, 2004.

[11] Rivest R. L. `http://theory.lcs.mit.edu/ rivest/chaffing.txt`.

[12] Lamport L. Constructing digital signatures from a one way function, 1979.

[13] Katz J., Schröder D., and Yerukhimovich A. Impossibility of blind signatures from one-way permutations. In Yuval Ishai, editor, *TCC*, volume 6597 of *Lecture Notes in Computer Science*, pages 615–629. Springer, 2011.

[14] Cichoń J., Klonowski M., and Kutyłowski M. Privacy protection for RFID with hidden subset identifiers. In *Pervasive*, pages 298–314, 2008.

[15] Golebiewski Z., Majcher K., and Zagórski F. Attacks on ckk family of rfid authentication protocols. In *ADHOC-NOW*, pages 241–250, 2008.

[16] Moskowitz P. A., Lauris A., and Morris S. S. A privacy-enhancing radio frequency identification tag: Implementation of the clipped tag. In *PerCom Workshops* [**45**], pages 348–351.

[17] Cichon J., Klonowski M., and Kutylowski M. Privacy protection in dynamic systems based on rfid tags. In *PerCom Workshops* [**45**], pages 235–240.

[18] Eschenauer L. and Gligor V. D. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on Computer and communications security 2002*, pages 41–47.

[19] Chan H., Perrig A., and Song D. Random key predistribution schemes for sensor networks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy 2003*, pages 197–213.

[20] Çamtepe S. A. and Yener B. Combinatorial design of key distribution mechanisms for wireless sensor networks. In Pierangela Samarati, Peter Y. A. Ryan, Dieter Gollmann, and Refik Molva, editors, *ESORICS*, volume 3193 of *Lecture Notes in Computer Science*, pages 293–308. Springer, 2004.

[21] Klonowski M., Kutylowski M., Ren M., and Rybarczyk K. Forward-secure key evolution in wireless sensor networks. In Feng Bao, San Ling, Tatsuaki Okamoto, Huaxiong Wang, and Chaoping Xing, editors, *CANS*, volume 4856 of *Lecture Notes in Computer Science*, pages 102–120. Springer, 2007.

[22] Juels A. and Weis S. A. *Authenticating Pervasive Devices with Human Protocols*, volume 3621. November 2005.

[23] Hopper N. J. and Blum M. Secure human identification protocols. *Lecture Notes in Computer Science*, 2248, 2001.

[24] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. On the inherent intractability of certain coding problems. In *IEEE Trans. Info. Theory*, pages 384–386, 1978.

[25] Håstad J. Some optimal inapproximability results. In *STOC*, pages 1–10, 1997.

[26] Levieil É. and Fouque P. A. An improved lpn algorithm. In Roberto De Prisco and Moti Yung, editors, *SCN*, volume 4116 of *Lecture Notes in Computer Science*, pages 348–359. Springer, 2006.

[27] Lyubashevsky V. The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem. In *APPROX-RANDOM* [**46**], pages 378–389.

[28] Blum A., Kalai A., and Wasserman H. Noise-tolerant learning, the parity problem, and the statistical query model. In *Journal of the ACM, vol. 50, no. 4*, pages 506–519, 2003.

[29] Katz J. and Shin J. S. Parallel and concurrent security of the hb and hb^+ protocols. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 73–87. Springer, 2006.

[30] Katz J. Efficient cryptographic protocols based on the hardness of learning parity with noise. In Steven D. Galbraith, editor, *IMA Int. Conf.*, volume 4887 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2007.

[31] Gilbert H., Sibert H., and Robshaw M. An active attack against a provably secure lightweight authentication protocol. In *IEEE Electronic Letters 41*, pages 1169–1170, 2005.

[32] Golebiewski Z., Majcher K., Zagórski F., and Zawada M. Practical attacks on hb and hb+ protocols. In Claudio Agostino Ardagna and Jianying Zhou, editors, *WISTP*, volume 6633 of *Lecture Notes in Computer Science*, pages 244–253. Springer, 2011.

[33] Gilbert H., Robshaw M. J. B., and Seurin Y. Good variants of hb^+ are hard to find. In Gene Tsudik, editor, *Financial Cryptography*, volume 5143 of *Lecture Notes in Computer Science*, pages 156–170. Springer, 2008.

[34] Gilbert H., Robshaw M. J. B., and Seurin Y. $Hb^{\#}$: Increasing the security and efficiency of hb^+. In *EUROCRYPT*, volume 4965 of *Lecture Notes in Computer Science*, pages 361–378. Springer, 2008.

[35] Ouafi K., Overbeck R., and Vaudenay S. On the security of hb# against a man-in-the-middle attack. In Josef Pieprzyk, editor, *ASIACRYPT*, volume 5350 of *Lecture Notes in Computer Science*, pages 108–124. Springer, 2008.

[36] Hammouri G. and Sunar B. Puf-hb: A tamper-resilient hb based authentication protocol. In Steven M. Bellovin, Rosario Gennaro, Angelos D. Keromytis, and Moti Yung, editors, *ACNS*, volume 5037 of *Lecture Notes in Computer Science*, pages 346–365, 2008.

[37] Bringer J., Chabanne H., Kevenaar T. A. M., and Kindarji B. Extending match-on-card to local biometric identification. In *COST 2101/2102 Conference*, volume 5707 of *Lecture Notes in Computer Science*, pages 178–186. Springer, 2009.

[38] Krawczyk H. Lfsr-based hashing and authentication. In Yvo Desmedt, editor, *CRYPTO*, volume 839 of *Lecture Notes in Computer Science*, pages 129–139. Springer, 1994.

[39] Dmitry Frumkin and Adi Shamir. Un-trusted-hb: Security vulnerabilities of trusted-hb. Cryptology ePrint Archive, Report 2009/044, 2009.

[40] Bringer J., Chabanne H., and Dottax E. Hb^{++}: a lightweight authentication protocol secure against some attacks. In *SecPerU*, pages 28–33. IEEE Computer Society, 2006.

[41] Piramuthu S. Hb and related lightweight authentication protocols for secure rfid tag/reader authentication. Proceedings of the Conference on Collaborative Electronic Commerce Technology and Research (CollECTer Europe), pp. 239-247, 2006.

[42] Munilla J. and Peinado A. Hb-mp: A further step in the hb-family of lightweight authentication protocols. *Comput. Netw.*, 51(9):2262–2267, 2007.

[43] Krause M. and Stegemann D. More on the security of linear rfid authentication protocols. In Michael J. Jacobson Jr., Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *Selected Areas in Cryptography*, volume 5867 of *Lecture Notes in Computer Science*, pages 182–196. Springer, 2009.

[44] Holcomb D. E., Burleson W. P., and Fu K. Power-up sram state as an identifying fingerprint and source of true random numbers. *IEEE Trans. Computers*, 58(9):1198–1210, 2009.

[45] *Fifth Annual IEEE International Conference on Pervasive Computing and Communications - Workshops (PerCom Workshops 2007), 19-23 March 2007, White Plains, New York, USA*. IEEE Computer Society, 2007.

[46] *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2005 and 9th InternationalWorkshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22-24, 2005, Proceedings*, volume 3624 of *Lecture Notes in Computer Science*. Springer, 2005.