# Readers Behaving Badly
## Reader Revocation in PKI-Based RFID Systems

Rishab Nithyanand, Gene Tsudik, and Ersin Uzun

Computer Science Department
University of California
Irvine, CA 92697
{rishabn,gts,euzun}@ics.uci.edu

**Abstract.** Recent emergence of RFID tags capable of performing public key operations motivates new RFID applications, including electronic travel documents, identification cards and payment instruments. In this context, public key certificates form the cornerstone of the overall system security. In this paper, we argue that one of the prominent challenges is how to handle revocation and expiration checking of RFID reader certificates. This is an important issue considering that these high-end RFID tags are geared for applications such as e-documents and contactless payment instruments. Furthermore, the problem is unique to public key-based RFID systems, since a passive RFID tag has no clock and thus cannot use (time-based) off-line methods.

In this paper, we address the problem of reader certificate expiration and revocation in PKI-Based RFID systems. We begin by observing an important distinguishing feature of *personal* RFID tags used in authentication, access control or payment applications – the involvement of a human user. We take advantage of the user's awareness and presence to construct a simple, efficient, secure and (most importantly) feasible solution. We evaluate the usability and practical security of our solution via user studies and discuss its feasibility.

## 1 Introduction

Radio Frequency Identification (RFID) is a wireless technology mainly used for identification of various types of objects, e.g, merchandise. An RFID tag is a passive device, i.e., it has no power source of its own. Information stored on an RFID tag can be read by special devices called RFID readers, from some distance away and without requiring line-of-sight alignment. Although RFID technology was initially envisaged as a replacement for barcodes in supply chain and inventory management, its many advantages have greatly broadened the scope of possible applications. Current and emerging applications range from visible and personal (e.g., toll transponders, passports, credit cards, access badges, livestock/pet tracking devices) to stealthy tags in merchandise (e.g., clothes, pharmaceuticals and library books). The cost and capabilities of an RFID tag vary widely depending on the target application. At the high end of the spectrum are

D. Gritzalis, B. Preneel, and M. Theoharidou (Eds.): ESORICS 2010, LNCS 6345, pp. 19–36, 2010.

the tags used in e-Passports, electronic ID (e-ID) Cards, e-Licenses, and contactless payment instruments. Such applications involve relatively sophisticated tags each costing a few (usually $< 10$ ) dollars. These tags are powerful enough to perform public key cryptographic operations.

In the "real world", one of the main security issues in using public key cryptography is certificate revocation. Any certificate-based public key infrastructure (PKI) needs an effective revocation mechanism. Revocation can be handled implicitly, via certificate expiration, or explicitly, via revocation status checking. Most PKI-s use a combination of implicit and explicit methods. The latter can be done off-line, using Certificate Revocation Lists (CRLs) [12] and similar structures, or on-line, using protocols such as Open Certificate Status Protocol (OCSP) [27]. However, as discussed below, these approaches are untenable in public key-enabled RFID systems.

Intuitively, certificate revocation in RFID systems should concern two entities: RFID tags and RFID readers. The former only becomes relevant if each tag has a "public key identity". We claim that revocation of RFID tags is a non-issue, since, once a tag identifies itself to a reader, the latter (as the entity performing a revocation check) can use any current revocation method, except perhaps OCSP which requires full-time Internet connectivity. This is reasonable because an RFID reader is a full-blown computing device with its own clock as well as ample power, memory, secondary storage and communication interfaces. Consequently, it can avail itself of any suitable revocation checking technique.

In contrast, revocation of readers is a problem in any public key-enabled RFID system. While a tag may or may not have public key identity, a reader must have one; otherwise, the use of public key cryptography becomes non-sensical. Therefore, before a tag discloses any information to a reader, it must make sure that the reader's public key certificate (PKC) is not expired or revoked.

### 1.1 Why Bother?

One common and central purpose of all RFID tags and systems is to enable tag identification (at various levels of granularity) by readers. With that in mind, many protocols have been proposed to protect the identification process (i.e., the tag-reader dialog) from a range of attacks. In systems where tags can not perform cryptographic operations or where they are limited to symmetric cryptography, reader revocation is not an issue, since it is essentially impossible. Whereas, in the context of public key-enabled tags, reader revocation is both imperative and possible, as we show later in this paper. It is imperative, because not doing it prompts some serious threats. For example, consider the following events: a reader is *lost*, *stolen*, *compromised* (perhaps without its owner's knowledge), or *decommissioned*.

In all of these cases, if it cannot be revoked effectively, a reader that has fallen into the wrong hands can be used to identify and track tags. In case of personal tags – e.g., ePassports, credit-cards or eIDs – other threats are possible, such as identity theft or credit card fraud.

Thus far, it might seem that our motivation is based solely on the need to detect *explicitly revoked* reader certificates[1]. However, what if a reader certificate naturally expires? This indicates *implicit revocation* and a well-behaved reader would not be operated further until a new certificate is obtained. However, if a reader (or rather its owner) is not well-behaved, it might continue operation with an expired certificate. Without checking certificate expiration, an unsuspecting tag could be tricked into identifying itself and possibly divulging other sensitive information.

In the remainder of this paper, we make no distinction between explicit revocation (i.e., revocation before expiration) and implicit revocation (i.e., certificate expiration) checking. The reason is that both tasks are essential for security and both require current time.

### 1.2 Why Is Reader Revocation Hard?

When presented with a PKC of a reader, a tag needs to check three things: (1) *signature* of the issuing certification authority (CA), (2) *expiration* and (3) *revocation status*.

The first is easy for any public key-enabled (pk-enabled) tag and has been already incorporated into some reader authentication schemes [6], [14]. However, (2) and (3) are problematic. Note that even a high-end tag is a passive device lacking a clock. Thus, a tag, by itself, has no means of deciding whether a presented certificate is expired.

Revocation checking is even more challenging. First, similar to expiration, offline revocation checking (e.g., CRL-based) requires current time because the tag needs to check the timeliness of the presented proof of non-revocation. Also, communicating a proof of non-revocation entails extra bandwidth from the reader to the tag. For CRLs, the bandwidth is $O(n)$ and, for more efficient CRTs, the bandwidth is $O(log\ n)$ – a non-negligible number for large values of $n$, where $n$ is the number of revoked readers[2]. Whereas, online revocation checking protocols (such as OSCP) offer constant-size proofs of non-revocation. However, such protocols are unsuitable due to their connectivity and availability requirements ;see Section 3 for further discussion.

### 1.3 Roadmap

We focus on a class of pk-enabled RFID systems where tags are both personal and attended. This includes e-Passports, e-Licenses and contactless credit cards. *Personal* means that a tag belongs to a human user and *attended* means that a tag is supposed to be activated only with that user's (owner's) consent. Our approach is based on several observations:

[1] "Explicitly" means before the expiration of the PKC.

[2] The problem of the high communication cost of CRL-s in current solutions has been noted by Blundo, et al. [4].

- User/owner presence and (implicit) consent are already required for the tag to be activated.
- Low-cost and low-power flexible display technology is a reality, e.g., e-paper and OLED. In fact, passive RFID tags with small (6-10 digit) displays have been demonstrated and are currently feasible.
- Since certificate revocation and expiration granularity is usually relatively coarse-grained (i.e., days or weeks, but not seconds or minutes), users can distinguish between timely and stale date/time values.

The rest is straight-forward: a display-equipped tag receives, from a reader, a PKC along with a signed and time-stamped proof of non-revocation. After verifying the respective signatures on the reader's PKC and the non-revocation proof, the tag displays the lesser of: (1) PKC expiration time and (2) non-revocation proof expiration time. The user, who is assumed to be reasonably aware of current time, validates the timeliness of the displayed time. If it is deemed to be stale, the user aborts the interaction with the reader. Otherwise, user allows the interaction to proceed.

Organization: We summarize related work in Section 2 and overview some trivial solutions in Section 3. We describe our approach in Section 4, followed by results of the usability study in Section 5. The paper ends with the summary in Section 6.

## 2 Related Work

There are many ways of handling certificate revocation. Of these, Certificate Revocation Lists (CRLs) are the most commonly used mechanism. Notably, CRLs are used by the X.509 Public Key Infrastructure for the Internet [12]. Some techniques improve the efficiency of revocation checking. Certificate Revocation Trees (CRTs) [19] use Merkle's Hash Trees [23] to communicate a relatively short non-revocation proofs (of size log n). Skip-lists [9] and 2-3 Trees [28] improve on the CRT update procedure through the use of dynamic data structures, offering asymptotically shorter proofs. Online Certificate Status Protocol (OCSP) [27] is an on-line method that reduces storage requirements and provides timely revocation status information. Certificate Revocation System (CRS) [25,24] offers fully implicit certificate revocation by placing the bulk of revocation burden on the prover (certificate owner) and yields compact proofs of certificate validity.

In spite of substantial prior work in both certificate revocation and RFID security, very little has been done with respect to reader revocation and expiration checking. However, the problem has been recognized in previous literature [26,11,15,10,7,30].

## 3 Trivial Solutions

We now consider some trivial reader revocation techniques and discuss their shortcomings.

### 3.1 Date Register and Time Stamps

Every PKC has a validity period defined by its effective date ($D_{eff}$) and expiration date ($D_{exp}$). During certificate verification, a tag can use the date stored in its register ($D_{curr}$) to determine whether a certificate has expired. Verification steps are as follows:

1. Tag verifies the CA signature of the reader's certificate.
2. Tag checks that $D_{exp}$ is greater than $D_{curr}$.
3. If (1) and (2) succeed, the tag accepts the certificate. If $D_{eff}$ is greater than $D_{curr}$, the tag updates $D_{curr}$ to $D_{eff}$.

With this approach, the estimate of the current date – $D_{curr}$ – stored by the tag is not guaranteed to be accurate and thus can not always protect it from readers with expired or revoked certificates. This is especially the case for a tag that has not been used for some time. The value of $D_{curr}$ might reflect a date far in the past, exposing the tag to attacks from readers revoked at any point after $D_{curr}$.

### 3.2 On-Line Revocation Checking

Online revocation-checking approaches, such as OCSP [27], alleviate client storage requirements by introducing trusted third parties (responders) that provide on-demand and up-to-date certificate status information. To validate a certificate, a client sends an OCSP status request to the appropriate responder and receives a signed status. In its basic form, OCSP requires a clock on the client, as it uses time-stamps to assure freshness. However, an optional OCSP extension supports the use of nonces as an alternative.

Although suitable for a large and well-connected infrastructure, such as a private network or the Internet, OCSP is problematic in RFID systems. Its use would require a tag to generate random challenges and conduct a 2-round (online) challenge-response protocol with an OCSP responder. Random challenges must be generated using a Pseudo-Random Number Generator (PRNG), which requires extra resources on the tag. More importantly, OCSP would necessitate constant infrastructure connectivity for all readers and availability of OCSP responders. Furthermore, the turnaround time for tag-reader interaction would become dependent on external factors, such as congestion of the communication infrastructure (e.g., the Internet) and current load on OCSP responders. Either factor might occasionally cause significant delays and prompt the need for back-up actions.

### 3.3 Internal Clocks

An internal clock would allow tags to accurately determine whether a certificate is expired and whether a non-revocation proof is current. However, a typical RFID tag is a purely passive device powered by radio waves emitted from a nearby reader. Since a real-time clock needs uninterrupted power, it cannot be sustained by passive tags. One might consider equipping RFID tags with batteries, however, this raises a slew of new problems, such as battery cost, clock synchronization and battery replacement.

## 4 Proposed Technique

We re-emphasize that our approach is aimed only at pk-based RFID systems. It has one simple goal: secure and reliable revocation checking on RFID tags. In the rest of this section, we discuss our assumptions and details of the proposed solution.

### 4.1 Assumptions

Our design entails the following assumptions[3]:

1. Each tag is owned and physically attended by a person who understands tag operation and who is reasonably aware of the current date.
2. Each tag is equipped with a one-line alpha-numeric (OLED or ePaper) display capable of showing a 6-8 digit date.
3. Each tag has a mechanism that allows it to become temporarily inaccessible to the reader (i.e., to be "turned off").
4. Each tag is aware of the name and the public key of a system-wide trusted certification authority (CA).
5. The CA is assumed to be infallible: anything signed by the CA is guaranteed to be genuine and error-free.
6. The CA issues an updated revocation structure (e.g., a CRL) periodically. It includes serial numbers of all revoked reader certificates.
7. Each tag knows the periodicity of revocation issuance (i.e., it can calculate the expiration date of revocation status information by knowing its issuance date.)
8. While powered up by a reader, a tag is capable of maintaining a count-down timer.
9. A tag can retain (in its non-volatile storage) the last valid date it encountered.
10. [**Optional**] A tag may have *a single button* for user input.

### 4.2 Basic Idea

Before providing any information to the reader, a tag has to validate the reader PKC. Recall our assumption that the user is physically near (e.g., holds) his tag during the entire process. Verification is done as follows:

1. The freshly powered-up tag receives the CRL and the reader PKC. Let $CRL_{iss}$, $CRL_{exp}$, $PKC_{iss}$ and $PKC_{exp}$ denote issuance and expiration times for purported CRL and PKC, respectively. Let the last valid date stored in the tag be $Tag_{Curr}$.

[3] Although we use "date" as the revocation/expiration granularity, proposed technique is equally applicable to both coarser- and finer-granular measures, e.g., month or hour.


**Fig. 1.** A Display and Button Equipped RFID Tag

2. If either $CRL_{exp}$ or $PKC_{exp}$ is smaller than $Tag_{curr}$, or $CRL_{iss} \geq PKC_{exp}$, the tag aborts.
3. The tag checks whether the CRL includes the serial number of the reader certificate. If so, it aborts.
4. The tag checks the CA signatures of the PKC and CRL. If either check fails, the tag aborts.
5. If $CRL_{iss}$ or $PKC_{iss}$ is more recent than the currently stored date, the tag updates it to the more recent of the two.
6. The tag displays the lesser of the $CRL_{exp}$ and $PKC_{exp}$. It then enters a countdown stage of fixed duration (e.g., 10 seconds).
7. The user views the date on the display.
   [**OPTION A:**]
   (a) If the displayed date is not in the past, the user does nothing and interaction between the tag and the reader resumes after the countdown stage.
   (b) Otherwise, the user terminates the protocol by initiating an escape action while the tag is still in countdown stage.
   [**OPTION B:**] (If Assumption 10 holds)
   (a) If the displayed date is in the future, the user presses the button on the tag before the timer runs out, and communication with the reader continues normally.
   (b) Otherwise, the timer runs out and the tag automatically aborts the protocol.

### 4.3 Escape Actions

As evident from the above, an escape action is required whenever the user decides that the displayed date is stale. Although the choice of an escape action is likely to be application-dependent, we sketch out several simple and viable examples.

**Using a Button:** Recent developments in low-power hardware integration on contactless cards have led to deployment of buttons on RFID tags [20,33]. On such tags, the user can be asked to press a button (within a fixed interval) as a signal of acceptance[4]. If the button is not pressed within that interval, the protocol is automatically terminated by the tag. Thus, the escape action in this case involves no explicit action by the user. We recommend this variant over alternatives discussed below, since it complies with the *safe defaults* design principle, i.e., without explicit approval by the user, the tag automatically aborts its interaction with the reader.

**Faraday Cages:** A Faraday Cage is a jacket made of highly conductive material that blocks external electric fields from reaching the device it encloses. Since tags are powered by the electric field emitted from a reader, it is theoretically possible to isolate them from all reader access by simply enclosing them in a Faraday cage. For tags that have an enclosing Faraday cage – such as e-Passports that have one inside their cover pages – the natural escape action is simply closing the passport.

**Disconnecting Antennas:** An RFID tag communicates and receives power through a coil antenna attached to its chip. Disconnecting the antenna from the chip immediately halts communication and shuts down the tag. A simple physical switch placed between a tag and its antenna can be used as an escape action. Similar mechanical actions aimed to halt communication between a tag and a reader are described in [17]. One drawback of such techniques is that physical damage to the tag is possible if the switch is handled roughly.

### 4.4 Efficient Revocation Checking

Although we hinted at using CRLs earlier in the paper, our approach would work with CRTs or any other off-line revocation scheme. However, both CRLs and CRTs become inefficient as the number of revoked readers increases. CRLs are linear and CRTs – logarithmic, in the number of revoked certificates. Our goal is to minimize bandwidth consumed by revocation information by making it constant, i.e, $O(1)$. To achieve this, we take advantage of a previously proposed modified CRL technique originally intended to provide privacy-preserving revocation checking [29].

In traditional CRLs, the only signature is computed over the hash of the entire list of revoked PKCs. Consequently, the entire list must be communicated to the verifier. To make CRLs bandwidth-optimal, [29] requires the CA or a Revocation Authority to sign each (sorted) entry in a CRL individually and bind it with the previous entry. In more detail, the modified CRL technique works as follows: assume that the CRL is sorted in ascending order by the revoked certificate serial

[4] For tags that have no buttons but built-in accelerometers, gestures (see [8] for more details) can also be used to signal user acceptance.

numbers. For a CRL with $n$ entries, the CA generates a signature for the $i$-th entry ($1 < i \leq n$) as follows:

$$Sign(i) = \{h(CRL_{iss}||SN_i||SN_{i-1})\}_{SK_{RA}}$$

where, $CRL_{iss}$ is the issuance date of this current CRL, $SN_i$ is the $i$-th certificate serial number on the ordered CRL, $SN_{i-1}$ is the immediately preceding revoked serial number, $SK_{RA}$ is the secret key of the CA and $h$ is a suitable cryptographic hash function. To mark the beginning and the end of a CRL, the CA uses two fixed sentinel values: $+\infty$ and $-\infty$.

When authenticating to a tag, a non-revoked reader provides its own PKC as well as the following constant-size non-revocation proof:

$$SN_j,\ SN_{j-1},\ CRL_{iss},\ Sign(j)$$

where reader certificate serial number $SN_{rdr}$ is such that $SN_{j-1} < SN_{rdr} < SN_j$. The reader PKC, along with the above information, allows the tag to easily check that: (1) the range between adjacent revoked certificate serial numbers contains the serial number of the reader PKC, and (2) the signature $Sign(j)$ is valid. If both are true, the tag continues with the protocol by displaying the lesser of the $CRL_{exp}$ and $PKC_{exp}$, as in step 6 of Section 4.2.

Compared with traditional CRLs, this scheme reduces both storage and communication overhead from $O(n)$ to $O(1)$ for both, readers and tags. On the other hand, the CA has to separately sign each CRL entry. Although this translates into significantly higher computational overhead for the CA, we note that CAs are powerful entities running on resource-rich systems and CRLs are not usually re-issued very frequently, i.e., weekly or daily, but not every minute or even every hour.

### 4.5 Security Considerations

Assuming that all cryptographic primitives used in the system are secure and the user executes necessary escape actions in case of expired (or revoked) reader certificates, the security of the proposed reader revocation checking mechanism is evident.

We acknowledge that user's awareness of time and ability to abort the protocol (when needed) are crucial for the overall security. To this end, we conducted some usability studies, including both surveys and experiments with a mock implementation. As discussed in section 5, our studies showed that people are reasonably aware of date and also able to execute the protocol with low error rates.

### 4.6 Cost Assessment

Recent technological advances have enabled mass production of small inexpensive displays (e.g., ePaper) that can be easily powered by high-end RFID tags

aided by nearby readers[5]. The current (total) cost of an ePaper display-equipped and public key-enabled RFID tag is about 17 Euros in quantities of 100,000 and the cost goes down appreciably in larger quantities [33]. Although this might seem high, we anticipate that the cost of cutting-edge passive display technologies (i.e., ePaper and OLED) will sharply decrease in the near future. Moreover, once a display is available, it can be used for other purposes, thus amortizing the expense. We briefly describe some potential alternative uses for display-equipped RFID tags:

**Transaction Verification:** RFID tags are commonly used as payment and transaction instruments (e.g., credit, ATM and voting cards). In such settings, a direct auxiliary channel between the tag and the user is necessary to verify the details of a transaction. This problem becomes especially apparent with payment applications. A malicious reader can easily fool the tag into signing or authorizing a transaction for an amount different from that communicated to the user. A display on a contactless payment card would solve this problem by showing the transaction amount requested by the reader on its display and waiting for explicit user approval before authorizing it.

**Device Pairing:** A display may be used for secure pairing of tags with other devices that do not share a CA with the tag. Visual channel-based secure device pairing methods that are proposed for personal gadgets can be used with display-equipped RFID tags (See [21] and [18] for a survey of such methods). The ability to establish a secure ad-hoc connection with arbitrary devices is a new concept for RFID tags that might open doors for new applications, e.g., the use of NFC-capable personal devices (e.g., cell-phones) to change and control settings on personal RFID tags.

**User/Owner Authentication:** In some scenarios, it might be necessary for a user to authenticate to a tag (e.g., credit card or passport). Currently this can be done only via trusted third party devices such as readers, mobile phones [31], personal computers and wearable beepers [16]. However, in the future, with a display-equipped RFID tag, the need for additional trusted devices might be obviated.

## 5 Usability

Since the proposed technique requires active user involvement, its usability is one of the key factors influencing its potential acceptance. Also, due to the nature of the protocol, certain type of user errors (i.e., accepting an incorrect or stale date) can result in a loss of security. Thus, we conducted two separate usability studies: online surveys and hands-on usability experiments. The goal of these studies was to answer the following questions:

1. Do everyday users worry about the reader revocation problem?

[5] Power feasibility analysis of integrating a display into a passive RFID tag circuit is discussed in Appendix A.

2. How do these users rate the usability of our solution?
3. Are users reasonably aware of the current date? What are the expected error rates?

### 5.1 Usability Experiment

In order to assess the usability of our method in the context of real users, 25 subjects were recruited to take part in the usability study. In order to prevent subjects from being explicitly aware of the date during the tests, care was taken to avoid setting up prior test appointments. Instead, subjects were recruited by the test coordinator at various campus venues, e.g., cafés, dorms, classrooms, offices, labs and other similar settings.

**Apparatus and Implementation:** Our test mock-up was implemented using two mobile phones: a Nokia N95 [2] (simulating the tag) and a Nokia E51 [1] (simulating the reader). These devices were chosen since, at the time of this study, actual RFID tags with displays and buttons could not be ordered in modest quantities. We used *Bluetooth* as the wireless communication medium between the N95 and E51. All implementation code was written in Java Mobile Edition. The time period for the automatic reject was set to 10 seconds.

**Subjects:** Our study participants were mainly students at the University of California, Irvine. Their age was well distributed among three groups: 36% – 18-24, 32% – 25-29, 32% – 30 +. Gender distribution was controlled for and almost evenly split between male and female (52% and 48%, respectively). On the other hand, 80% of the subjects had a bachelors degree, thus yielding a rather educated sample. We attribute this to the specifics of the study venue (a university campus).

**Procedure:** To help subjects in understanding the concept of personal RFID tags, the ePassport example was used throughout the test and the questionnaire phases. First, subjects were asked not to consult any source of current date/time before and during the tests. Then, they were given a brief overview of our method and the importance of maintaining natural behavior during the experiments. Next, each subject was presented with a mock-up implementation and was asked to execute the protocol six times. Finally, subject opinions were solicited via the post-test questionnaire.

The set of dates used in the study process was: +/-1 day, -3 days, +7 days, -29 days, and -364 days, from the actual test date (Note that "+" and "-" indicate future and past dates, respectively). All experiments were conducted during the first week of December 2009, and choices of -29 days and -364 days were deliberate so as to make the staleness of these dates more deceiving to the subjects.

Test cases were presented to each subject in random order. The test administrator held the phone simulating the reader and sent dates to the device simulated the tag. After a date was displayed on the "personal tag", the test subject was asked to decide whether to: (1) accept the date by pressing the

button within ten seconds, or (2) reject it by doing nothing. The process was repeated six times for each test-case.

**Results**
*Completion Time and Error Rates:* For subjects who accepted displayed dates, the study yielded average completion time of 3.07 seconds, with standard deviation of 1.58 seconds. This shows that subjects were quick in reacting whenever they considered the date to be valid. This also confirms that our choice of a 10-second time-out was appropriate.

Among the 25 subjects, the false negative rate (reject for a date that was not stale) was quite low. No one rejected a date that was one day in future, and only one subject (4% of the sample) rejected the date that was seven days in the future. The false positive rate (accept a stale date) was also low in all cases, except one. When subjects were shown dates that were, respectively: 1, 3 and 29 days earlier, the corresponding observed error rates were 0%, 0% and 4%. However, surprisingly, the error rate spiked up to 40% when subjects were shown a date that was almost a year (364 days) earlier. We discuss this further in Section 5.3 below.

*User Opinions:* Subjects who tried our mock-up implementation rated its usability at 77% on the original System Usability Scale (SUS) [5], a score that is about 13% higher than that obtained from the on-line survey, where participants rated it solely based on its written description. 84% of the subjects who tested our implementation stated that they would like this system implemented on their own personal tags, while 12% were neutral to the idea (the average score on a 5-point Likert scale was 4.1 with the standard deviation of 0.75).

### 5.2 On-Line Survey

We created an online survey [3] that was used to anonymously sample 98 individuals. The purpose was to collect information regarding perceived usability and general acceptance of our solution, rather than its actual usability. Participants were given an explanation of the reader revocation problem. Then, they were presented with the detailed description of our approach that included all user interaction.

**Survey Results:** The proposed technique yielded a score of 68/100 on the system usability scale (SUS). 66% of the participants stated that they would like to see it implemented on their E-passports, while 26% were neutral (the average score on 5-point Likert scale was 3.67 with the standard deviation of 0.87). 84% of the participants were worried about identity theft and 88% stated that they are concerned about revealing personal information to unauthorized parties in general.

In the online survey, we did not ask the subjects for their estimate of the current date or whether a displayed is stale, as this data would have been severely biased owing to the availability of the current date on their computer screen. Instead, participants were asked about their general awareness of the current

date. 40% indicated that they are usually aware of the exact date, 35% were confident to know it with at most one-day error margin and 22% claimed to be within the +/- 3-day range. The remaining 3% indicated that 7 or more days error would be possible on their estimate of the current date.

### 5.3 Discussion

Based on our usability results, we now attempt to answer the questions raised at the beginning of this section:

*Are people concerned with the problem we aim to solve?* Among the 123 total participants (98+25, in both studies) 88% are worried about revealing information to unauthorized parties. 70% said that they wanted to see the proposed technique implemented on their personal tags.

*How do people rate the usability of our approach?* Given the detailed description of the method and required interaction, 98 participants rated its usability at 68% on SUS scale. The usability rating was even higher (77%) for 25 subjects who actually experimented with the mock-up implementation. Both scores are above industry averages [22] and indicate good usability and acceptability characteristics.

*Are users aware of current date?* As results show, our method very rarely yields false negatives: users are capable of not mistaking valid (future) dates for being in the past. As far as false positives, however, results are mixed. Stale days and months are, for the most part, easily recognized as such. However, with the stale year, the error rate is quite high, at 40%. This deserves a closer look. While we do not claim to know the exact reason(s), some conjectures can be made.

When confronted with a date, most of us are conditioned to first check day and month, e.g., current dates on documents and expiration dates on perishable products. At the same time, users do not tend to pay as much attention to more gross or blatant errors (such as wrong year) perhaps because they consider it to be an unlikely event. Also, we note that among six test-cases for each user, just one had a date with the wrong year. This may have inadvertently conditioned the subjects to pay more attention to the month/day fields of the dates.

On the other hand, we anticipate that year mismatches will be quite rare in practice, since the tags can record the most recent *valid* date they encounter. Therefore, dates with stale year values will be mostly automatically detected and rejected by tags without the need for any user interaction. However, high user error rates in wrong year values can still pose a threat if a tag is not used for a year or longer.

Another approach that *may* yield lower error rates is showing *today's date* to the users instead of an expiration date. This approach can be implemented as follows:

1. The reader sends the tag its claimed value for "today's date" ($D_{curr}$) in addition to its PKC and the most recent CRL.

2. The tag checks that $D_{curr} < PKC_{exp}$ and $D_{curr} < CRL_{exp}$. If either check fails, the tag aborts.
3. The tag displays $D_{curr}$ to the user.
4. The user is now required to verify that the displayed date is indeed "today's date".

We believe more comprehensive user-studies are needed to evaluate whether the above approach or certain changes in date representation and formatting (for e.g., displaying YYYY/MM/DD instead of MM/DD/YY) might help lower user errors.

## 6 Conclusions

In this paper, we presented a simple and effective method for reader revocation on pk-enabled RFID tags. Our approach requires each tag to be equipped with a small display and be attended by a human user during certificate validation. As long as the user (tag owner) plays its part correctly, our solution eliminates the period of vulnerability with respect to detecting revoked readers.

Recent advances in display technology, such as ePaper and OLED, have already yielded inexpensive display-equipped RFID tags. The low cost of these displays combined with the better security properties and potential new application domains make displays on RFID tags a near reality. Moreover, our usability studies suggest that users find this solution usable and they are capable of performing their roles within reasonable error rates. We believe that display-equipped RFID tags will soon be in mass production and the method proposed in this paper will be applicable to a wide variety of public key-enabled tags.

**Acknowledgments.** The authors are grateful to Bruno Crispo and Markus Ullman for their valuable comments on the previous version of this paper. This work is supported in part by NSF Cybertrust grant #0831526.

## References

1. Nokia e51 specifications, `http://europe.nokia.com/find-products/devices/nokia-e51/specifications`
2. Nokia n95 specifications, `http://www.nokiausa.com/find-products/phones/nokia-n95-8gb/specifications`
3. Display enabled identification and payment instruments (November 2009), `http://sprout.ics.uci.edu/projects/usec/survey.html`
4. Blundo, C., Persiano, G., Sadeghi, A.-R., Visconti, I.: Resettable and Non-Transferable Chip Authentication for ePassports. In: Conference on RFID Security (2008)
5. Brooke, J.: Sus - a quick and dirty usability scale. Usability Evaluation in Industry (1996)
6. Bundesamt fur Sicherheit in der Informationstechnik. Advanced Security Mechanisms for Machine Readable Travel Documents : Version 2.0 (2008)

7. Cheon, J.H., Hong, J., Tsudik, G.: Reducing RFID Reader Load with the Meet-in-the-Middle Strategy. Cryptology ePrint Archive, Report 2009/092 (2009)
8. Czeskis, A., Koscher, K., Smith, J.R., Kohno, T.: Rfids and secret handshakes: defending against ghost-and-leech attacks and unauthorized reads with context-aware communications. In: Computer and Communications Security – CCS (2008)
9. Goodrich, M., Tamassia, R.: Efficient authenticated dictionaries with skip lists and commutative hashing, US Patent App. 10/416,015 (May 7, 2003)
10. Heydt-Benjamin, T., Bailey, D., Fu, K., Juels, A., O'hare, T.: Vulnerabilities in first-generation RFID-enabled credit cards. Financial Cryptography and Data Security (2007)
11. Hoepman, J.-H., Hubbers, E., Jacobs, B., Oostdijk, M., Wichers Schreur, R.: Crossing Borders: Security and Privacy Issues of the European e-Passport. In: Yoshiura, H., Sakurai, K., Rannenberg, K., Murayama, Y., Kawamura, S.-i. (eds.) IWSEC 2006. LNCS, vol. 4266, pp. 152–167. Springer, Heidelberg (2006)
12. Housley, R., Ford, W., Polk, W., Solo, D.: RFC 2459: Internet X.509 public key infrastructure certificate and CRL profile (January 1999)
13. Infineon Technologies AG, AIM CC. Preliminary Short Product Information: Chip Card and Security IC's (2006)
14. International Civil Aviation Organization. Machine Readable Travel Documents: Specifications for Electronically Enabled Passports with Biometric Identification Capability (2006)
15. Juels, A., Molnar, D., Wagner, D.: Security and privacy issues in e-passports. In: Security and Privacy for Emerging Areas in Communications Networks – SECURECOMM (2005)
16. Kaliski, B.: Future directions in user authentication. In: IT-DEFENSE (2005)
17. Karjoth, G., Moskowitz, P.A.: Disabling rfid tags with visible confirmation: clipped tags are silenced. In: Workshop on Privacy in the Electronic Society – WPES (2005)
18. Kobsa, A., Sonawalla, R., Tsudik, G., Uzun, E., Wang, Y.: Serial hook-ups: a comparative usability study of secure device pairing methods. In: Symposium on Usable Privacy and Security – SOUPS (2009)
19. Kocher, P.C.: On certificate revocation and validation. In: Hirschfeld, R. (ed.) FC 1998. LNCS, vol. 1465, pp. 172–177. Springer, Heidelberg (1998)
20. Kugler, D., Ullman, M.: Contactless security tokens - enhanced security by using new hardware features in cryptographic based security mechanisms. In: Dagstuhl Seminar Proceedings of Foundations for Forgery - Resilient Cryptographic Hardware (July 2009)
21. Kumar, A., Saxena, N., Tsudik, G., Uzun, E.: Caveat eptor: A comparative study of secure device pairing methods (2009)
22. Lewis, J., Sauro, J.: The factor structure of the system usability scale. In: Proceedings of the Human Computer Interaction International Conference (HCII 2009), San Diego CA, USA (2009)
23. Merkle, R.C.: Secrecy, authentication, and public key systems. Technical report, Stanford University (1979)
24. Micali, S.: Efficient certificate revocation. Technical Memo MIT/LCS/TM-542b, Massachusetts Institute of Technology (1996)
25. Micali, S.: Certificate revocation system. United States Patent, US Patent 5,666,416 (September 1997)
26. Monnerat, J., Vaudenay, S., Vuagnoux, M.: About Machine-Readable Travel Documents. In: Conference on RFID Security (2007)
27. Myers, M., Ankney, R., Malpani, A., Galperin, S., Adams, C.: Internet public key infrastructure online certificate status protocol- ocsp (1999)

28. Naor, M., Nissim, K.: Certificate revocation and certificate update. Technical report (1999)
29. Narasimha, M., Solis, J., Tsudik, G.: Privacy preserving revocation checking. International Journal of Information Security 8(1), 61–75 (2009)
30. Oren, Y., Feldhofer, M.: A Low-Resource Public-Key Identification Scheme for RFID Tags and Sensor Nodes. In: ACM Conference on Wireless Network Security – WiSec (2009)
31. Saxena, N., Uddin, M. B., Voris, J.: Treat 'em like other devices: user authentication of multiple personal rfid tags. In: SOUPS (2009)
32. Scholz, P., Reihold, C., John, W., Hilleringmann, U.: Analysis of energy transmission for inductive coupled rfid tags. In: International Conference on RFID (2007)
33. Ullman, M.: Personal communication (September 2009)

## A Power Feasibility Analysis

The aim of this section is to show that it is completely feasible to integrate low power display technologies on passive RFID tags without any change on reader specifications. We analyze the maximum power requirements of the proposed system and its effect on the (theoretical) maximum working distance with current readers. In the rest of this section, we use ePassports as an example due to their clear tag and reader specifications.

We propose the use of display technologies such as ePaper, OLED, and other such low-power bistable displays. These displays require power of the order of 100mW (for a 2" display unit) during display updates and 0mW of power during standby.

### A.1 Power Analysis

ePassport tags such as those supplied by Infineon Technologies, require up to 55mW of power to operate [13] while the display unit requires a maximum power of 100mW to operate. We analyze the power requirements of the proposed system from three aspects:

1. The ePassport tag is operating at maximum power and the display unit is static or non-existent.
2. The ePassport tag is on standby and the display unit is being updated (*i.e.*, refreshed).
3. The ePassport tag is operating at maximum power and the display unit is being updated (*i.e.*, refreshed).

In the first case, the power required by the ePassport circuit to operate will be $\sim$ 55mW (the power required by the display unit at this time is zero). In the second case, the power required by the ePassport circuit to operate will be $\sim$100mW (the power required by the tag during standby is negligible). In the final case, the power required by the ePassport circuit to operate will be $\sim$155mW (the sum of the maximum power required by the tag and display).

The ePassport tag and reader when placed parallel to each other can be represented as a circuit, with circuit parameters set in the manner described by Scholz *et al.* [32].

First, we establish a relationship between the mutual inductance ($M$) and the distance ($x$) between the antenna of the tag and the reader.

$$M = \frac{\mu\pi N_1 N_2 (r_1 r_2)^2}{2\sqrt{(r_1^2 + x^2)^3}} \tag{1}$$

Where $\mu$ is the Permeability $[H/m]$; $N_1$ and $N_2$ are the number of turns in the antennas of the tag and reader; $r_1$ and $r_2$ are the radii $[mm]$ of each of these turns. Substituting default values [32] we get the relation

$$M = \frac{1.57 \times 10^{-12}}{x^3} \tag{2}$$

Now we establish a relationship between the power required by the tag ($P_{Tag}$) and distance ($x$). This is done through the series of equations below.

$$P_{Tag} = I_1^2 R_T \tag{3}$$

Where $I_1$ is the current running in the reader circuit $[mA]$ and $R_T$ represents the tag impedance which is given by (4).

$$R_T = \frac{M^2 R_L}{L_2^2} \tag{4}$$

Where $L_2$ is assigned a value of 168nH [32] and $R_L$ is the load resistance given by (5).

$$R_L = \frac{V_T^2}{P_{Tag}} \tag{5}$$

$V_T$ is the voltage required in the tag circuit (5.5 Volts). The value of $R_L$ is 195.1 $\Omega$ in the case that the ePassport tag and display unit operate at maximum power together (case 3). $R_L$ is 302.5 $\Omega$ in the case that the ePassport tag is on standby when the display unit is refreshed (case 1). Finally, by combining equations 2 through 5, we can get a relationship between $x$ and $P_{Tag}$.

$$x^6 = \frac{(1.57 \times 10^{-12})^2 \times (I_1)^2 \times (R_L)}{P_{Tag} \times (L_2)^2} \tag{6}$$

Making the necessary substitutions, we get the following values for $x$, where $x$ represents the maximum possible operating distance:

– An ePassport tag without a display unit or with display on stand-by (*i.e.*, not refreshing):

$$P_{Tag} = 55\ mW,\ R_L = 550\ \Omega \Longrightarrow x = .097\ m \tag{7}$$

– An ePassport display unit while refreshing output when the tag is in standby mode:

$$P_{Tag} = 100\ mW,\ R_L = 302.5\ \Omega \Longrightarrow x = .080\ m \tag{8}$$

– An ePassport tag and the display unit operating at maximum power:

$$P_{Tag} = 155\ mW,\ R_L = 195.1\ \Omega \Longrightarrow x = .069\ m \tag{9}$$

From the above results it is clear that even with the current reader and antenna specification, adding a display reduces the maximum operating distance between the tag and reader only by 2.8 cm. Therefore, adding a display unit to the current ePassport circuit is feasible and doesn't require any changes over the power specifications in the original proposal [6]. If longer operating distances (over 6.9 cm) are needed, it can be achieved with small modifications on the RFID antenna design or by increasing power of a reader.