Identification-Free Batch Authentication for RFID Tags

Lei Yang*‡, Jinsong Han*‡, Yong Qi*, Yunhao Liu†‡,
young@srfid.org, hjs.jason@mail.xjtu.edu.cn, qiy@mail.xjtu.edu.cn, liu@cse.ust.hk
* Department of Computer Science and Technology, Xi'an Jiaotong University, China
† TNLIST, School of Software, Tsinghua University, China
‡ Department of Computer Science and Engineering, HKUST, Hong Kong

***Abstract*—Cardinality estimation and tag authentication are two major issues in large-scale Radio Frequency Identification (RFID) systems. While there exist both per-tag and probabilistic approaches for the cardinality estimation, the RFID-oriented authentication protocols are mainly per-tag based: the reader authenticates one tag at each time. For a batch of tags, current RFID systems have to identify them and then authenticate each tag sequentially, incurring large volume of authentication data and huge communication cost. We study the RFID batch authentication issue and propose the first probabilistic approach, termed as Single Echo based Batch Authentication (SEBA), to meet the requirement of prompt and reliable batch authentications in large scale RFID applications, *e.g.*, the anti-counterfeiting solution. Without the need of identifying tags, SEBA provides a provable probabilistic guarantee that the percentage of potential counterfeit products is under the user-defined threshold. The experimental result demonstrates the effectiveness of SEBA in fast batch authentications and significant improvement compared to existing approaches.**

***Index Terms*—RFID, Batch Authentication, Identification-free, Anti-Counterfeiting, SEBA**

I. Introduction

Radio Frequency IDentification (RFID) is a promising technique and widely employed in a variety of applications, such as logistic and supply chain management [1], access control [2], theft prevention [3], and movement tracking [4]–[6]. A RFID system usually consists of a large number of tags and readers. The RFID tags are typically low-cost and pervasive devices, being attached to products or targets to enable the identification of those objects. A tag has small microchips and an antenna on board. The reader can collect the IDs of tags via RF signals, without the need of keeping in sight or touch. As an effective automatic processing measures, RFID offers several attractive features over the barcode, such as non-optical proximity, interactive communication, rewritable ability, and *etc.*

An emerging RFID application is anti-counterfeiting [7], *i.e.*, to verify the authenticity of the products. Counterfeiting behavior is considered as one of the most serious threats to the economy. Recently, the counterfeit goods account for 5% of the world trade, which involves billions of US dollars every year [8]. Many RFID enabled anti-counterfeiting solutions have been introduced in logistics, retailing, passports, banknotes, *etc.* Compared to other anti-counterfeiting technologies, RFID anti-counterfeiting has a great advantage that it enables efficient and automatic verification, especially in the case of massive products.

A common way of RFID enabled anti-counterfeiting is that the manufacture stores a serial number k (or termed as *key*) for each tag. The serial number will also be stored in an authentication server maintained by the manufacture. During authentication, the customer uses a RFID reader to obtain the serial number from the tag and sends this number to an authentication server. If the serial number is valid, the product to which the tag is attached is declared as genuine. During this process, however, as the wireless RF channel is open, an eavesdropper can easily overhear the serial number and create a counterfeit tag. To address this issue, many efforts have been made for designing more efficient and private authentication protocols. Weis *et al.* [9] propose a hash function based authentication scheme, Hash Lock. In Hash Lock, the reader sends a random number r as the authentication request. The tag generates a hash value on the inputs of r and k. The hash value is replied to the reader for authentication. If the authentication server can find a key that is able to generate a same hash value with r, the tag is verified. The search complexity of Hash Lock is $\mathcal{O}(N)$, where N is the total number of tags in the system. To improve the search efficiency, tree-based approaches [10]–[12] convert the verification process to a Depth-First-Search in a balanced key tree to reduce the search complexity to $\mathcal{O}(\log(N))$. As a result, a tree based protocol requires each tag to store $\mathcal{O}(\log(N))$ keys, and the tag and reader have to exchange $\mathcal{O}(\log(N))$ hash values for authentication. Although tree based approaches reduce the search complexity, the size of authentication data increases an order of magnitude so that the reader needs much longer scanning time to obtain data.

Indeed, existing approaches are impractical to authenticate a large number of tags, where the quantity of tags can be up to tens of thousands. We observe three bottlenecks that seriously affect the authentication efficiency. (1) Large scanning time. To authenticate a batch of tags, the reader must employ an anti-collision algorithm to identify them before obtaining their hash values. The efficiency of identification in most anti-collision algorithms, however, is very low. (2) High volume of authentication data. The length of hash values is 20 bytes if using SHA-1 hash function. Authenticating a batch of tags needs to transfer $20 * n * \log(N)$ bytes data, where n denotes the number of tags in the batch. (3) Significant server workload. Due to the high volume of authentication data, huge communication traffic might be incurred in the system server. This will insert significant workload on the server and further aggravates the authentication delay. For example, given $N = 1,000,000$ tags in the system and $n = 10,000$ tags in each batch. According to the well-known RFID standard ISO-18000, the average identification throughput is about 100 tags per second [13], [14]. Therefore, the time for identifying one batch of tags will be $10,000/100 \approx 100s$. The authentication data transferred from tags to the reader is about $20 * 10^4 * \log_2(10^6) \approx 3.8M$ bytes. Since every bit requires $25\mu s$ for transmission [9], the reader will spend around $3.8M * 25\mu s \approx 13$ minutes to collect the authentication data from the batch in an ideal case, even if we ignore the synchronization process and retransmission caused by signal loss or interference. In addition to the traffic delay in the server, authenticating the batch of products totally consumes over $100/60 + 13 \approx 14$ minutes. Therefor, one reader can only uninterruptedly verify $24 * 60/14 \approx 103$ batches per day! Obviously, such an extremely low authentication efficiency is unacceptable in practice.

By reconsidering the solution of batch authentication in another perspective, we find it is not always necessary to ensure the genuineness of every single product in a batch. The fact is, even in the genuine products, there might be some defective ones shipped from the manufacture. It is acceptable if we guarantee the percentage of counterfeit products is sufficiently small. To this end, we propose the first approach to verify the validity of a batch of tags. The naive solution for verifying the validity of a batch of tags is to sample some tags from the batch and authenticate them one by one. However, as we will analyze in Section III, Sampling based Batch Authentication (SMP) scheme performs poorly in breaking the above three bottlenecks. For example, we still need to authenticate 30% products in each batch to guarantee the percentage of potential counterfeit products is under 0.3% with a high confidence 99.99%, where the amount of authentication data is up to $1.1M$ bytes.

To solve the problem of batch authentication, we design a Single Echo based Batch Authentication (SEBA) protocol. In order to fully utilize different encoding modes of the authentication data, we further present two variant versions, SEBA-2 and SEBA-3. The most distinct feature of SEBA is identification-free, *i.e.*, it does not need to identify any tag before authentication. SEBA provides a provable probabilistic guarantee for valid batches of tags that the percentage of potential counterfeit products is less than ε with a high probability $1 - \delta$. Concurrently, we reduce 88% scanning time and 95.2% communication cost compared to the SMP design.

The rest of this paper is organized as follows. We introduce preliminary knowledge about RFID systems in Section II and discuss the Sampling based Batch Authentication protocol in Section III. Our Single Echo Batch Authentication protocol is presented in Section IV. In Section V, we examine the performance of SEBA with simulations based on real traces and extend the evaluation to a simulated large-scale system. At last, we review the related works in Section VI and conclude this paper in Section VII.

II. Preminary

We first briefly discuss the Framed Slotted ALOHA anti-collision protocols and tree-based authentication protocol, and then introduce our system model and formulate the problem of batch authentication.

A. Framed Slotted ALOHA Protocol

Framed Slotted ALOHA (FSA) is a popular anti-collision protocol adopted by mainstream RFID organizations and manufactures. The design of our protocols is partially based upon FSA. Using FSA to identify a batch of tags, the reader first divides a detecting frame into f slots and broadcasts the f before the identification procedure. Each tag contains a pseudo random number generator $h(x)$, which is used to choose the slot number. After receiving f, each tag selects $h(ID)\ mod\ f$, $h_f(ID)$ for short, as its slot number. The reader then sequentially scans every slot in the frame. The reader uses a 'slot start' command to start a slot. In each slot, if a tag's slot number equals zero, it will send its ID to the server immediately. Otherwise, the tag reduces its slot number by one. Since a tag cannot sense the signals replied from other tags, there are three types of slots from the reader's perspective. If none of tags responds in a slot, the slot is termed as *idle slot*. If only one tag responds in a slot, the reader can successfully receive the tag's ID. Accordingly, such a slot is termed as *single slot*. If there are more than one tag responding in a slot, the slot is termed as *collided slot*. At the end of frame, if collisions have occurred in this frame, the reader will start a new frame until all tags are identified. Given n tags in a batch and the frame length f, the probability of the i^{th} slot is an idle slot, single slot, or collided slot can be computed as follows:

$$P_0(n, f) = (1 - \frac{1}{f})^n \approx \exp(-\frac{n}{f})$$

$$P_1(n, f) = \binom{n}{1}(\frac{1}{f})(1 - \frac{1}{f})^{n-1} \approx \frac{n}{f}\exp(-\frac{n}{f})$$

$$\begin{aligned} P_X(n, f) &= 1 - P_0(n, f) - P_1(n, f) \\ &\approx 1 - \exp(-\tfrac{n}{f}) - (\tfrac{n}{f})\exp(-\tfrac{n}{f}) \end{aligned}$$

B. Tree based Authentication

Existing tree based approaches [10]–[12] construct a balanced tree to organize the keys for all tags. In the key tree, each node stores a key and each tag is arranged to a leaf node. The keys in the path from the root to a leaf node are assigned to the tag that is related to the leaf node. For the example illustrated in Figure 1, tag T_5 has keys $k_0, k_{1,2}, k_{2,3}, k_{3,5}$. When the reader authenticates T_5, it sends a nonce r to T_5. T_5 computes hash values of $h(r, k_0)$, $h(r, k_{1,2})$, $h(r, k_{2,3})$, and $h(r, k_{3,5})$, and sends them to the reader in sequence. After receiving the response, the reader searches appropriate keys in the key tree to locate the tag. The procedure is equivalent to finding a path from the root to the leaf node assigned to T_5. If such a path exists, T_5 is a valid tag. In the above procedure,

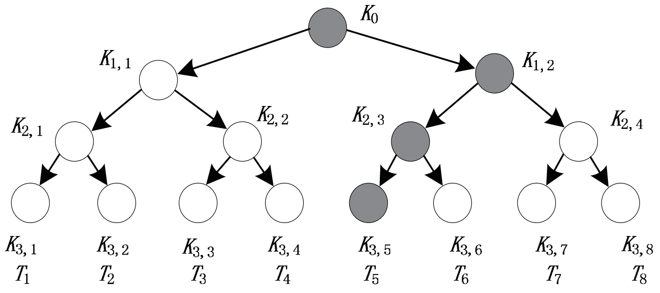

Fig. 1. Authentication Procedure of Tree based Approaches

each tag must transfer 4 hash values to the reader at each authentication. As we discussed before, such a large volume of data is a major bottleneck preventing us from accelerating the batch authentication. Note that our protocols are completely compatible with tree based approaches and only refer to the keys located in leaf nodes.

C. System Model and Problem Formulation

In our model, an RFID system contains three components: an authentication server, a number of readers, and batches of tags. The authentication server maintains N keys in database and has powerful computing capability. A reader connects to the authentication server through a high speed network, for example the Internet. We assume that each product is attached with an RFID tag. The number of tags in a batch, denoted as n, is known in advance, or we can online query the value of n from the database. The communication model between the reader and tag is based on slotted wireless channels, in which an interaction between the reader and tags is conducted within predefined and equally spaced intervals, called *slots*. The reader guarantees the slot synchronization via energizing probes/requests. Compared to the duration of each slot, the delay produced by the wireless channel is negligible.

Each tag contains a unique key k. We call a batch of tags is valid if none of counterfeit tag is *detected*, otherwise it is invalid. Our goal is to quickly and accurately determine the validity of a given batch of tags. Ideally, we can say a batch is valid only if we successfully authenticate every tag in the batch. However, such a deterministic approach is difficult to perform due to the time consuming identification and authentication procedures. In this paper, we design the first probabilistic protocols to solve the batch authentication problem. We use two parameters to depict the probabilistic features: tolerance ε and confidence level $1-\delta$, both of which are given by the user in advance. We guarantee a batch is valid with probability greater than $1-\delta$ if there are no more than $n * \varepsilon$ counterfeit tags in the batch. Note that it does not mean that the batch will be declared as valid if the number of counterfeit tags is lower than $n * \varepsilon$. Even if there is only one counterfeit tag in the batch, the batch will still be declared as invalid as long as we successfully detect any counterfeit tag. For probabilistic protocols, if the fraction of counterfeits in one batch is really lower than the tolerance, we cannot guarantee to detect the counterfeits with the given confidence, but it is still possible to detect them.

Anti-counterfeiting is important in many applications. For example, the logistics companies, retailing enterprisers, and customs, have the need of fast validating a batch of products to confirm whether the products are genuine before processing them. Instead of performing time-consuming per-tag authentication, our scheme enables prompt validation on a batch of products with high probability to determine whether there are counterfeits in the batch. If the result is true, the application can directly accept these products as valid. Otherwise, the applications can refuse the products or perform the per-tag authentication on the batch to filter the counterfeits.

III. Sampling Based Batch Authentication

In this section, we outline the SaMPling based Batch Authentication (SMP) protocol, and consider it as a benchmark for the design of our protocols.

A. Design

SMP consists of three steps: identification, sampling, and authentication. First, the reader identifies all tags in the batch via FSA in order to know which tags in the batch. Second, it randomly selects n_α tags as samples and collects the authentication data from them. Third, the reader forwards the data to the authentication server. If the authentication server finds one invalid tag, SMP can determine the batch is invalid. Due to the randomness of sampling, it is necessary to select enough samples to guarantee the probability that at least one of n_α counterfeit products is chosen is more than $1-\delta$.

B. Analysis

According to the sampling principle, we adopt samples without replacements in which one tag only appears at most once as a sample. Then the number of counterfeit products follows the hyper-geometric distribution. We define random variable Y to represent the number of counterfeit products in samples and α as the sampling ratio. The discrete probability density function of Y is given by [16]:

$$h(y; n\alpha, n\varepsilon, n) = \frac{\binom{n\varepsilon}{y}\binom{n(1-\varepsilon)}{n\alpha-y}}{\binom{n}{n\alpha}}$$

The expected value of Y is $E(Y) = n\alpha\varepsilon$. Since the probability that a given tag is selected multiple times is near to zero when the n is sufficiently larger, we consider Y approximates Possion Distribution with $\lambda = n\alpha\varepsilon$. Therefore, we have Theorem 1, which indicates that only when α is not smaller than $-\ln(\delta)/(n\varepsilon)$, SMP is able to sample counterfeit products from the batch with the probability greater than $1-\delta$. As the example given in the introduction, we set $\varepsilon = 0.003$ and $\delta = 0.0001$. According to Theorem 1, we need to sample about 30% products. The reader thereby needs to read $10^4 * 30\% * 20 * \log_2(10^6) = 1.1M$ bytes of authentication data from sampled tags. The corresponding overhead is relatively high.

Theorem 1: Given n, ε , and δ, if the batch includes counterfeit products, the probability that there is at least one counterfeit product being sampled is greater than $1-\delta$ iff the sample ratio $\alpha \geq -\ln(\delta)/(n\varepsilon)$.

Proof: The probability that there is at least one counterfeit product in the samples is given by:

$$\sum_{k=1}^{n\alpha} \Pr(Y=k) = 1 - \Pr(Y=0) = 1 - \exp(-\lambda) \geq 1 - \delta$$

Since $\lambda = n\varepsilon\alpha$, $\alpha \geq -\ln(\delta)/(n\varepsilon)$. ■

IV. Single Echo based Batch Authentication

In this section, we present the design of Single Echo based Batch Authentication (SEBA) protocol. We also analyze the performance and security of SEBA.

A. SEBA Design

SEBA comprises of three steps: authentication initialization, Echo Sketch (*ES*) retrieval, and *ES* authentication. The procedure is illustrated in Figure 2.

Authentication initialization: The reader launches an authentication request to the authentication server with three inputs, the number of tags in the batch n, the acceptable counterfeit ratio ε, and the failure probability δ. The server returns the frame length f with a nonce r to the reader. Determining the optimal length of frames is a crucial task for minimizing the latency of batch authentications in this step.

ES retrieval: The reader broadcasts the f and r received from the server in the first step. In SEBA, we refine the slot selecting mechanism used in FSA. We let each tag choose $h_f(k,r)$ as its slot number, in which the slot number selection is based on the tag's key instead of its ID. After the frame ends, the reader abstracts the responses in the frame as an *Echo Sketch (ES)*. *ES* is a vector, in which each element is related to a slot in the frame. There are three types of elements in an *ES*, 0, 1, and X, representing empty slot, single slot, and collided slot, respectively. For example, $ES = [0\ 1\ 0\ X\ 1\ X]$. The 1^{st} and 3^{rd} slots are empty slots, the 2^{nd} and 5^{th} slots are single slots, and the 4^{th} and 6^{th} sots are collided slots. We can consider an ES as the authentication fingerprint of a batch of tags when the length of ES is long enough. In our approach, every tag does not transfer its ID in the slot, but a short random signal (usually < 10 bits [1], [17]), as long as the reader can detect these signals. Therefore, the time duration of all slots in our approaches is very short. To differentiate from longer responses used by existing identification solutions, we call such a short response as an *echo*. Since one tag only responds once, we term our protocol as Single Echo based Batch Authentication (SEBA) protocol.

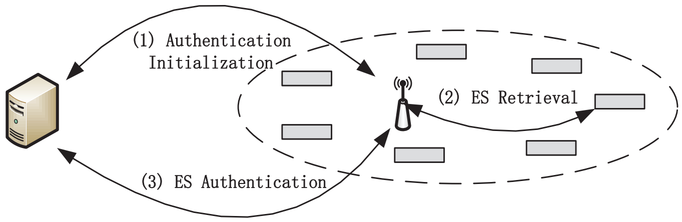

Fig. 2. Authentication Procedure of SEBA

TABLE I
Echo Sketch Union Operation

	0	1	X
0	0	1	X
1	1	X	X
X	X	X	X

ES authentication: The ES is forwarded to the authentication server by the reader for validation. Since the server stores all the keys of tags in the database, it can choose n keys from N keys for reconstructing the ES. If such reconstructed ES exists, the server deterministically accepts the batch of tags as valid. Otherwise, the batch is invalid. However, the reconstruction of ES is a time consuming process since there are $\binom{N}{n}$ combinations to reconstruct ES. On the other hand, our goal is not to find the genuine tags but determine whether there is any counterfeit tag. Thus, the batch authentication problem can be converted to a problem of detecting outlier echoes. Because a counterfeit tag has no valid key, its corresponding echo is not expected, like the report from an outlier. The detecting process can be sketched as follows. The server checks every element in *ES*. (1) If $ES[i] = 0$, the server directly moves to $ES[i+1]$. (2) If $ES[i] = 1$, the server tries to find a key set $K = \{k | h_f(k,r) = i\}$. If $|K| > 0$, the server moves to next element. Otherwise, if $|K| = 0$, which means none of genuine tag should emit echo in this slot, the server stops and asserts the existence of outlier echoes. (3) If $ES[i] = X$, the server finds a key set $K = \{k | h_f(k,r) = i\}$. If $|K| < 2$, which means at most one genuine tag exists but more than two tags emit echo in scenario, the server asserts the existence of outlier echoes. Otherwise, the server moves to next element. After the checking ends, if there is no outlier echo detected in the ES, the batch will be accepted as valid. We will prove the correctness and completeness next subsection.

B. Performance Analysis

We assume that each tag in the batch, including genuine and counterfeit tags, must reply once in the frame. This assumption seems to be strict, since the counterfeit tag may deliberately keep silent or emit multiple meaningless echoes. We call these two kinds of behaviors as the *hidden attack* and *stimulated attack*. We will discuss how to detect them in the next subsection.

1) Outlier Echoes Detecting: We formulate the problem of outlier echo detecting as follows. Since the slot number is randomly and independently chosen by each tag, we can consider the echo sketches received by the reader as a special union of two virtual echo sketches. The one echo sketch is produced by the echoes from $n(1-\varepsilon)$ genuine tags, denoted as ES_G; the other is produced by the echoes from $n\varepsilon$ counterfeit tags, denoted as ES_C. Namely, $ES = ES_G \cup ES_C$, where $\cup$ is a special union operation defined in Table I. For example, if $ES_G = [\ 0\ 0\ 0\ 1\ 1\ 1\ X\ X\ X]$ and $ES_C = [0\ 1\ X\ 0\ 1\ X\ 0\ 1\ X]$, then $ES = ES_G \cup ES_C = [0\ 1\ X\ 1\ X\ X\ X\ X\ X]$. In theory, the probability of the i^{th} slot is an empty slot, single slot, or collided slot in ES_C can be represented as follows:

$$\Pr(ES_C[i]=0)=P_0(n\varepsilon,f)$$

$$\Pr(ES_C[i]=1)=P_1(n\varepsilon,f)$$

$$\Pr(ES_C[i]=X)=P_X(n\varepsilon,f)$$

Not all the outlier echoes from counterfeit tags can be detected by the server via the received echo sketches, because the echoes from genuine tags may conceal the echoes from counterfeit tags. For example, if $ES_G[i]=X$ and $ES_C[i]=1$, then $ES[i]=X$. In this case, the server has no idea whether there is an outlier echo from counterfeit tags. However, if the frame is sufficiently long, the outlier echoes can be eventually detected.

The detecting procedure can be abstract as comparison between two echo sketches. One is the ES received by the reader and the other is a virtual global echo sketch ES_U produced by the echoes from all genuine tags, with the same f and r as those of *ES*. We list the comparison between them as follows:

C1:$ES_U[i]=0, ES[i]=0$: There should be no any genuine tag choosing this slot. We cannot employ such a result to determine whether there are counterfeit tags in the batch.

C2:$ES_U[i]=0, ES[i]=1$: There should be no any genuine tag selecting this slot. But the result shows that one tag in the batch chooses this slot. We can ensure that this tag must be a counterfeit.

C3:$ES_U[i]=0, ES[i]=X$: There should be no any genuine tag selecting this slot. But the result shows that more than one tag chooses this slot. We can ensure that they must be counterfeit tags.

C4:$ES_U[i]=1, ES[i]=0$: There should be at most one genuine tag choosing this slot. But the result shows that no tag in the batch chooses the slot. We cannot ensure whether there are counterfeit tags.

C5:$ES_U[i]=1, ES[i]=1$: There should be at most one genuine tag choosing this slot. The result also shows that one tag chooses the slot. We cannot ensure whether this tag is a counterfeit.

C6:$ES_U[i]=1, ES[i]=X$: There should be at most one genuine tag choosing this slot. But the result shows that more than one tag chooses the slot. We can ensure that the batch must have counterfeit tags.

C7:$ES_U[i]=X, ES[i]=0$: There should be more than one genuine tag choosing this slot. But the result shows that no tag in the batch chooses the slot. We cannot ensure whether there are counterfeit tags.

C8:$ES_U[i]=X, ES[i]=1$: There should be more than one genuine tag choosing this slot. But the result shows that only one tag in the batch chooses the slot. We cannot ensure whether the tag is a counterfeit.

C9:$ES_U[i]=X, ES[i]=X$: There should be more than one genuine tag choosing this slot. The result also indicates that multiple genuine tags choose this slot. We cannot ensure whether there are counterfeit tags.

In short, three cases, C2, C3, and C6, can be used for outlier echo detecting. In C2, $ES_U[i]=0, ES[i]=1$. It is equivalent to $ES_U[i]=0, ES_C[i]=1$, since the echo comes from one counterfeit tag. Therefore, the probability of C2 happening is given by:

$$\begin{aligned}\Pr(C2)&=\Pr(ES_U[i]=0, ES_C[i]=1)\\&=P_0(N,f)P_1(n\varepsilon,f)\end{aligned}$$

Similarly, we have

$$\begin{aligned}\Pr(C3)&=\Pr(ES_U[i]=0, ES_C[i]=X)\\&=P_0(N,f)P_X(n\varepsilon,f)\end{aligned}$$

$$\begin{aligned}\Pr(C6)&=\Pr(ES_U[i]=1, ES_C[i]=X)\\&=P_1(N,f)P_X(n\varepsilon,f)\end{aligned}$$

Therefore, the probability that the i^{th} element is an outlier echo, termed as Q, is given by:

$$\begin{aligned}Q&=\Pr(C2)+\Pr(C3)+\Pr(C6)\\&=\frac{e^{-\frac{n\varepsilon+N}{f}}\left(f^2\left(e^{\frac{n\varepsilon}{f}}-1\right)+fN\left(e^{\frac{n\varepsilon}{f}}-1\right)-nN\varepsilon\right)}{f^2}\end{aligned}$$

Correspondingly, the total probability that our approach detects the counterfeit tags, termed as P, is given by:

$$\begin{aligned}P&=1-(1-Q)^f\\&=1-\left(1+\frac{e^{-\frac{n\varepsilon+N}{f}}\left(f^2+fN+nN\varepsilon\right)}{f^2}-\frac{e^{-\frac{N}{f}}(f+N)}{f}\right)^f\end{aligned}$$

Note that P is a function with four inputs N, n, f, and ε, *i.e.* $P(N,n,f,\varepsilon)$.

2) Optimal Frame Length: Solving the batch authentication problem can be defined as: given N_0, n_0, ε_0, and δ_0, finding the minimum cost of detecting the counterfeit tags with a high probability larger than $1-\delta_0$, if there are more than $n_0\varepsilon_0$ counterfeit tags in the batch of tags. The cost means the overhead in terms of scanning time and the volume of authentication data. Both of them are proportional to the frame length. Hence, there are two competing forces influencing f: using a short frame decreases the total cost, while using a long frame provides more chances to find outlier echoes according to Lemma 1. Thus, the cost minimization depends on the optimal frame length which satisfies two conditions:

$$f=min\{f|P(N_0,n_0,f,\varepsilon_0)\geq 1-\delta_0\} \quad (1)$$

$$\forall\varepsilon>\varepsilon_0, P(N_0,n_0,f_{min},\varepsilon)>1-\delta_0 \quad (2)$$

The f_0 that is subject to $P(N_0,n_0,f_0,\varepsilon_0)=1-\delta_0$ is the optimal frame length. We prove the correctness of this claim in Theorem 2.

Lemma 1: Given N_0, n_0, and ε_0, $P(N_0,n_0,f_1,\varepsilon_0)<P(N_0,n_0,f_2,\varepsilon_0)$ while $f_1<f_2$.

Lemma 2: Given N_0, n_0,and f_0, $P(N_0,n_0,f_0,\varepsilon_1)<P(N_0,n_0,f_0,\varepsilon_2)$ while $\varepsilon_1<\varepsilon_2$.

Due to the limited space, we omit the proofs of Lemma 1and Lemma 2. Intuitively, increasing the frame length will introduce more idle slots to ES, which incurs more chances to find counterfeit tags. It is also obvious that more counterfeit tags tend to yield higher probability of being detected.

Theorem 2: Given N_0, n_0, ε_0 and δ_0, f_0 is the optimal frame length if it satisfies $P(N_0, n_0, f_0, \varepsilon_0) = 1 - \delta_0$.

Proof: For any $f < f_0$, according to Lemma 1, $P(N_0, n_0, f, \varepsilon_0) < P(N_0, n_0, f_0, \varepsilon_0) = 1 - \delta_0$. Thus, f_0 is subject to condition (1). On the other hand, for any $\varepsilon > \varepsilon_0$, $P(N_0, n_0, f_0, \varepsilon) > P(N_0, n_0, f_0, \varepsilon_0)$, according to Lemma 2. Thus, f_0 is subject to condition (2). Hence, f_0 is the optimal frame length. ■

C. Security Analysis

We examine the potential attacks that can be launched to our protocol.

1) Stimulated and Hidden Attack: Since our approach depends on the tags' echoes instead of direct authentication, counterfeit tags can disturb the distribution of slots through two attacks, *Stimulated Attack* and *Hidden Attack*. In the stimulated attack, a counterfeit tag emits echoes in multiple slots for disturbing the distribution of slots in the ES. In fact, our approach is intrinsically immune to such attack, since generating more meaningless echoes is equivalent to increasing the ratio of counterfeit tags, which helps to increase the probability of detecting counterfeit tags based on Lemma 2. In the hidden attack, the counterfeit tags always keep silent to avoid the exposure. Defending against the hidden attack seems a little harder. However, we can utilize the fact that the hidden attack will increase the number of empty slots. According to Theorem 1 in [17], we can find that the number of empty slots M approximates a normal distribution, namely, $M \sim \mathcal{N}(\mu, \sigma^2)$, where $\mu = f\exp(-n/f)$, and $\sigma^2 = f\exp(-n/f)(1-(1+n/f)\exp(-n/f))$. Ultimately, we measure the instance of M, termed as m, and use it to detect hidden attacks. If $|m-\mu| \le \sigma$, none of hidden attack happens. Otherwise, if $|m - \mu| > \sigma$, the probability of hidden attack is above $1 - \frac{\sigma^2}{(m-\mu)^2}$ according to Chebyshev's inequality, *i.e.*, $\Pr(|M - \mu| \ge |m - \mu|) \le \frac{\sigma^2}{(m-\mu)^2}$. When the probability exceeds a threshold, we can assert the occurrence of hidden attack.

2) Eavesdropping and replay attacks: If any eavesdropper is within the coverage of a legal reader, it can easily capture the echoes between the reader and the genuine tags [18]–[20]. Even worse, the attacker can record the transmitted message between the legitimate reader and tag, and retransmit it to them later. Similar to previous protocols, SEBA employs random number r to defend against such attacks. Since the random number is generated uniformly at random for each authentication, it is extremely difficult for attackers to predetermine the number. In addition, the length of r in our approaches is sufficiently long (more than 64 bits) such that the probability of successfully guessing a random number is negligible. Thus, an attacker can neither disclose the secret information by overhearing the communication in SEBA nor leverage the replay attack to gain benefits.

3) Cloning Attack: As another effective attack, the adversary can steal the tags from genuine products, clone the stolen tags, and attach the cloned tags on the counterfeit products. These replicated tags are identical to the valid tags. The server cannot detect counterfeit products towards this attack. In the industry, manufactures usually employ special cover materials to wrap the tag such that the tag would be destroyed when the cover is opened. In addition, recent works [21] propose Physical UnClonable Functions (PUFs) that exploit the physical characteristics of the silicon to uniquely characterize each tag, which makes it impossible to clone a tag. Combining with those solutions, SEBA can resist the clone attack.

D. Discussion

1) Echo Sketch Encoding: SEBA needs two bits to encode each element for an ES, *i.e.*, 0, 1, and X. Totally, it needs $2f$ bits for encoding an entire ES. We call a SEBA using such an encoding mode as SEBA-3. For saving the storage overhead, we can merely use one bit to encode each element, in which both 1 and X are encoded as 1. If adopting this encoding mode, our approach is termed as SEBA-2. SEBA-2 reduces about 50% storage overhead from SEBA-3. However, since SEBA-2 misses the description of collided slots, we only use C2 case to detect the outlier echoes. Then the probability that the i^{th} element is an outlier echo becomes:

$$Q'(N, n, f, \varepsilon) = P_0(N, f)(P_1(n\varepsilon, f) + P_X(n\varepsilon, f))$$

It is obvious that $Q' < Q$. The total probability of SEBA-2 to detect the counterfeit tags is $P' = 1 - (1 - Q')^f$. This value is less than that of SEBA-3. Hence, SEBA-2 requires a much longer frame length to guarantee the same probability of detecting counterfeit tags as SEBA-3 in theory. We will further compare these two encoding modes in the evaluation section.

2) Resolution of the Optimal Frame Length: According to Theorem 2, one of our important tasks is to find the resolution of $P(N_0, n_0, f, \varepsilon_0) = 1 - \delta_0$. With this complex and implicit function, it is difficult to directly solve f. Fortunately, f is a non-negative integer and $P(N_0, n_0, f, \varepsilon_0)$ is an increasing function with f. Hence, we can find an approximate non-negative integer f that satisfies the follows inequality:

$$P(N_0, n_0, f, \varepsilon_0) \ge 1 - \delta_0 > P(N_0, n_0, f - 1, \varepsilon_0) \quad (3)$$

If we can estimate the lower bound and upper bound of f, we can quickly find the solution using binary search method.

The lower bound: Suppose that f is sufficiently small such that all echoes are collided with high probability, *i.e.*, 0.99, when n_0 tags are projected into ES, then such an ES is useless to detect any counterfeits. We consider the length of frame as the lower bound of f. Namely,

$$P_X(n_0, f_{lower}) = 0.99 \Rightarrow f_{lower} \approx \frac{1}{6}n_0$$

The upper bound: Suppose we have such an idea ES that it can even contain every echo from N_0 genuine tags and $n_0\varepsilon_0$ counterfeit without collisions, then any counterfeit echo can be exactly detected. The length of such an ideal ES can be estimated as the upper bound of echo sketch. Namely,

$$P_X(N_0, f_{upper}) = 0.01 \Rightarrow f_{upper} \approx 7(N_0 + n_0\varepsilon_0)$$

Thereby the search complexity equals $\mathcal{O}(\ln(N_0 + n_0\varepsilon_0))$.

V. Performance Evaluation

To examine performance of SMP, SEBA-2 and SEBA-3, we simulate SEBA design over real logistics traces collected from a processing center of global express mail service (EMS) provider. We also extend the dataset with more simulated data for evaluating the performance of SEBA in large-scale systems. Our evaluation focuses on three metrics: accuracy, scanning cost, and communication cost.

A. Test Setup and Methodology

Each day, the EMS provider deliveries $2,456$ mailing items through that center, such as the express mails, parcels, or pouches, to a medium-size city in average. Before the transportation, the mailing items with the same destination are encapsulated in one batch. The misdelivery, if happens, will incur tremendous cost, especially for those highly valuable items. Thus, the mails are authenticated their destinations with privacy preservation before delivery. The EMS provider can attach RFID tags to mailing items, and adopt authentication schemes to the tags. In our simulation, we compare our design with SMP and another 'Authentication All' (AA) approach. AA is a deterministic approach, which identifies and authenticates all tags for examining whether there is any counterfeit tag. Without using any samples, the accuracy of AA is 100% but the efficiency of AA is the worst. We adopt a recent balanced-binary-tree-based authentication protocol [10], as the single tag authentication method used by AA and SMP. We consider those mis-delivered tags as 'counterfeit' ones, although they are legal in system that they are supposed to be delivered.

We collect successful mail delivery records of a medium-size city within one month (December, 2009) as our basic dataset. It totally contains $78,606$ records, as shown in Figure 3. We take the quantity of daily mails as the size of a batch of tags for each day. For examining the delivery accuracy, we deliberately introduce $1\% \sim 3\%$ randomly generated counterfeits into the dataset. Since the amount of mails delivered to the city is up to $943,272$ every year. We set the depth of the key tree to 21 for supporting $N = 1,048,576$ tags. Correspondingly, each tag attached to a mail contains 21 keys in its memory. We adopt SHA-1 as the hash function. We set the tolerance $\varepsilon = 0.01$ and confidence $\delta = 0.05$.

In our simulation, the authentication server is implemented on a high performance PC, DELL PRECISION T3400, using Java as the programming tool. We adopt MySQL 5.1 as the database to store $2,097,152$ keys for the simulated tags. Each simulation takes 100 runs with the same parameters, and we report the average.

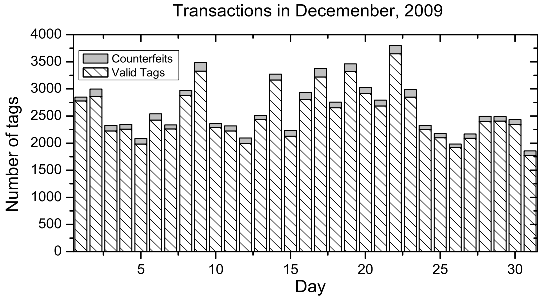

Fig. 3. Input dataset in our evaluation

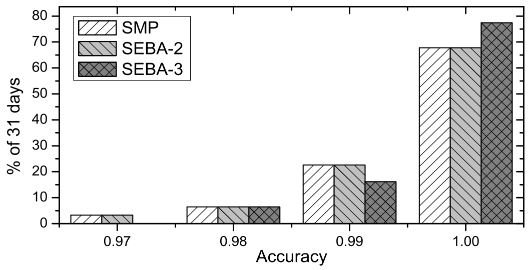

Fig. 4. Accuracy comparison among SMP, SEBA-2 and SEBA-3

B. Accuracy

We define the accuracy as the ratio of the times that the system correctly detects counterfeit tags to the total times in one day. This metric is related to the most crucial concern of the user on the counterfeit detection. To reflect the accuracy of the three approaches, we show the cumulative accuracy distribution in Figure 4. All three approaches offer above $1 - \delta = 0.95$ accuracy in practice. The mean accuracy of SEBA-3 is 0.997, while the mean accuracy of SMP and SEBA-2 is 0.995. In particular, SEBA-3 achieves accuracy of 1.0 in 24 days. On the other hand, both SMP and SEBA-2 achieve accuracy of 1.0 in 21 days, From the result, we can see that all of three probabilistic approaches fulfill our accuracy requirements.

C. Scanning Cost

Scanning cost reflects the time consumed for the interaction between the reader and tags. This metric is relevant to the key parameter that determines the processing speed of authentication and counterfeit detection.

Since every bit almost consumes the same transmission time which equals $25\mu s$ [15] on average, we measure the scanning cost by multiplying the size of transferred data (in terms of bits) with $25\mu s$. In SMP, the scanning cost contains two parts. One is generated by using FSA for identification. Since the length of ID is 96-bits [15], the total size of data used for identification equals $96 * f$. The second part is used for the tree-based authentication. Suppose that $n\alpha$ tags are sampled and the length of random numbers equals 64 bits. The size of second part equals $(160{*}21{+}64){*}n{*}\alpha$ bits. Therefore, the total size of SMP is given by $96 * f + (160 * 21 + 64) * n * \alpha$ bits. AA can be considered as a special sampling case in which $\alpha = 1.0$. On the contrary, the data transferred in SEBA-2 and SEBA-3 only contains one random number and f echoes. Since each echo is in the same size (10 bits), the total size of data transferred in SEBA-2 and SEBA-3 equals $64 + 10f$.

Figure 5 plots the scanning costs of AA, SMP, SEBA-2 and SEBA-3, respectively. We can find that the mean scanning cost of AA is $185.6s$. The scanning cost of SMP is $23.6s$, which is merely 12.7% of AA. Clearly, it shows that probabilistic

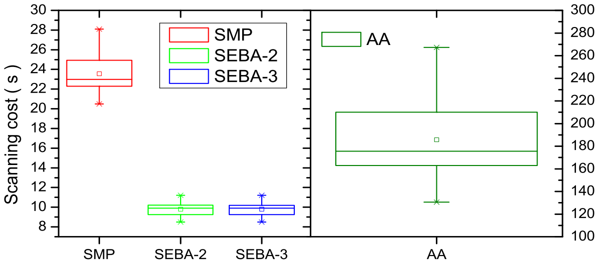

Fig. 5. Scanning costs (s) of AA, SMP, SEBA-2 and SEBA-3

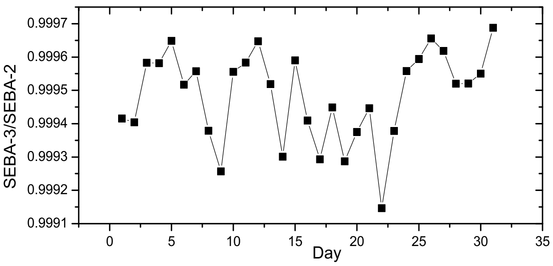

Fig. 6. The ratio of the scanning cost of SEBA-3 to that of SEBA-2

approaches significantly reduce the scanning cost. Moreover, the mean values of scanning costs of both SEBA-2 and SEBA-3 are $9.27s$, which provides almost 60% cost reduction from SMP. On the other hand, the cost variances of SEBA-2 and SEBA-3 are much lower than those of SMP or AA. The main reason is that both SMP and AA adopt FSA as the anti-collision algorithm. FSA is a probabilistic algorithm and its probabilistic feature may intensify the instability, besides the impact of variable sizes of batches. For SEBA-2 or SEBA-3, as long as keeping n, ε, and δ as constant, the cost is stable. The variance of SEBA-2 or SEBA-3 is mainly caused by the variance of the batch size.

As we analyze in Section IV-D1, the scanning cost of SEBA-2 should be always larger than that of SEBA-3 under the same parameter setting. Interestingly, the difference between them is small. Figure 6 shows the ratio of the scanning cost of SBEA-3 to that of SEBA-2. The protocols incur such similar costs that the ratio amplitude of them is only about 0.0008. In short, we can claim that the scanning costs of SBEA-2 and SEBA-3 are almost equivalent in practice.

D. Communication Cost

Communication cost is defined as the size of data transferred between the reader and authentication server. A low communication cost will alleviate the network traffic and workload on the server, and thereby improve the application performance. In SMP, the cost equals $160 * 21 * n * \alpha$ bits. SEBA-2 requires 1 bit and SEBA-3 require 2 bits to represent each element. Therefore, the cost of SEBA-2 and SEBA-3 equals f and $2f$, respectively.

Figure 7 shows the communication cost of the four approaches. SEBA-2 has the least communication cost, which is 0.5% of AA, 7.4% of SMP, and 50% of SEBA-3 on average.

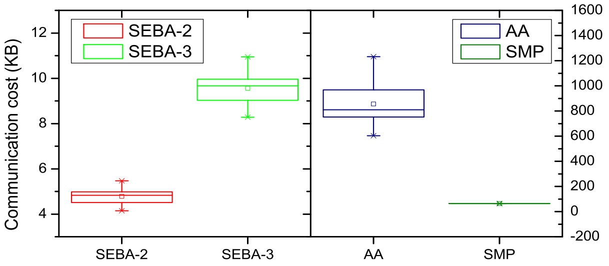

Fig. 7. Communication cost (KB) of AA, SMP, SEBA-2 and SEBA-3

The communication cost of SEBA-3, although doubles from SEBA-2, is only 1.1% and 14% of AA and SMP, which indicates that SEBA remarkably outperforms AA and SMP in terms of communication cost. Note that the communication cost of SMP collapses into a line in the figure. That is because the size of samples, $n_\alpha = n * \alpha = -\ln(\delta)/\varepsilon$, is not related to the size of the batch but only determined by the ε and δ, which are kept constant in our entire experiment.

E. Large-scale Simulations

We change the size of the batch, n, and examine the scalability of SEBA in large-scale systems. The size of the batch ranges from $1,000$ to $12,000$ genuine tags with a constant increment of $1,000$ tags. We also randomly generate $1\% \sim 3\%$ counterfeit tags to the batch. We set the tolerance and confidence level as 0.01 and 0.05, respectively.

We re-check the accuracy of SMP, SEBA-2 and SEBA-3 in variant sizes of batches. Figure 8 shows the accuracy distribution while the n varies from 1,000 to 12,000. All of the three approaches maintain high accuracy. Particularly, SEBA-3 is the most accurate one that achieves 100% detection of counterfeiting tags in about 75% cases. We find that the accuracy is lower than other cases when the sizes are 3000, 9000 and 10000. It is because the percentage of counterfeits is lower than 1.5% in these cases. As we discussed in Section IV, low percentage of counterfeits will leads to low probability of detecting counterfeits.

Improvement on the scanning cost is also studied. We show the result in Table II. We observe that as the batch size increases, the scanning cost of AA and SMP correspondingly increases. This is because the time consumed for delivering the data of identification and authentication is linear to the batch size. Although the size of samples keeps constant in our simulation, SMP still needs the identification process. Therefore, the cost increment of SMP derives from the augmentative time consumption in the identification phase. On the contrary, both the scanning costs of SEBA-2 and SEBA-3 are decreased. The reason is that enlarging batch size with constant counterfeits ratio magnifies the probability of detecting outlier echoes, while resulting in a length decline of the echo sketch. This fact indicates that SEBA performs much better than SMP or AA in large-scale scenarios. For example, when $n = 12,000$, AA needs 14 minutes and SMP needs 1 minute to detect the counterfeit tags, while SEBA-2 and SEBA-3 only spends 6 seconds.

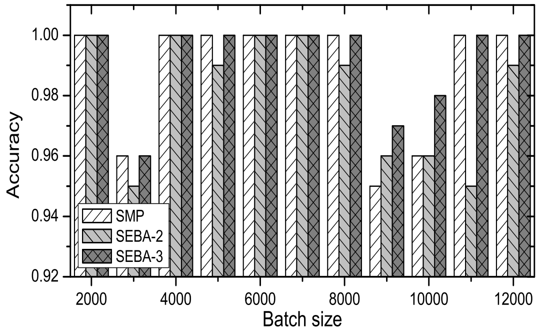

Fig. 8. Accuracy comparison in large-scale simulations

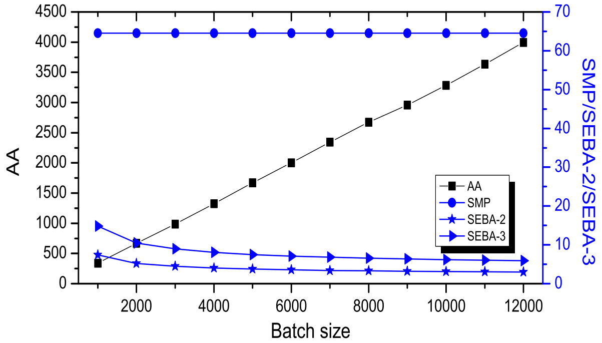

Fig. 9. Communication costs (KB) of AA, SMP, SEBA-2, and SEBA-3 in large-scale simulations

We also investigate the improvement of SEBA on the communication cost and show the result in Figure 9. As the batch becomes larger, the communication cost of SEBA-2 is only a small fraction of that of SMP (11% with 1,000 tags and only 4.6% with 12,000 tags). Compared to AA, SEBA-2 incurs very small communication cost (2.2% with 1,000 tags and only 0.07% with 12,000 tags of AA). SMP keeps the cost at round $64.5KB$ due to the fixed size of samples. Moreover, we can find that when the size of the batch is around 1000 tags, the difference seems not very large in terms of scanning time in Table II. However, the communication costs of SMP, SEBA-2, and SEBA-3 are 64.55KB, 14.8KB, and 7.411KB, when the number of tags equals 1,000. This observation indicates that although the three approaches do not have much difference in terms of scanning time when authenticating small-size batches, SEBA-2 and SEBA-3 can reduce a large volume of authentication data, and hence significantly improve the processing latency in the backend system, especially when the system comprises of many readers.

F. Selection of SEBA-2 and SEBA-3

From above observations, we can find that SEBA-3 is more accurate than SEBA-2. Both of them have similar scanning cost while SEBA-2 has 50% communication cost of SEBA-3. Selection of them should be application-specific. SEBA-3 seems a good choice if we emphasize the accuracy, while SEBA-2 should be chosen if we have strict latency limits or low network bandwidth between the reader and server.

VI. Related Works

The primary usage of RFID tags is to deterministically identify object or people via the attached tag. Collision is a critical problem in RFID systems when processing a batch of tags. In the literature, RFID anti-collision mechanisms comprise of two categories, Framed Slotted ALOHA (FSA) based [15], [22]–[24] and Binary Tree (BT) based [14], [25]–[27]. The well known RFID organization, EPC Global, adopts a variation of FSA, 'Q-Adaptive' in its protocol family, EPC Gen2 [15], which adaptively tunes the frame length according to the type of last slot. Lee *et al.*. [22] show that the FSA reader can obtain a maximum identification throughput within its scanning field when the size of detecting frame equals to the number of tags and propose a dynamic FSA for RFID systems. Sheng *et al.* [23] study a fundamental problem of continuous scanning in RFID systems and designs algorithms based on the information gathered in the previous scanning. Xie *et al.* [24] propose a probabilistic model, which involves the practical conditions of RFID systems, such as the path loss and multi-path effect, in their realistic settings to efficiently identify tags on the moving conveyor. The binary tree based algorithm has been adopted by another popular standard, ISO 18000-6 [14]. When designing tree based algorithms, researchers usually organize the tags with their IDs and identify the tags based on the query tree technique. Myung and Lee [25] propose an adaptive binary splitting (ABS) protocol to avoid collisions and efficiently identify tags based on the last identification.

Many approaches [10]–[12], [18], [28], [29] have been proposed to achieve private authentication in RFID systems. Weis *et al.* [28] propose a hash function based authentication scheme, Hash Lock, to prevent tags from being tracked. Hash Lock, being sufficiently secure, suffers from poor efficiency, as the complexity is *O(N)*, where N is the total number of tags. Subsequent approaches in the literature aims at reducing the cost of key search. Tree based structure can reduce the search complexity from $O(N)$ to $O(\log(N))$ [10]–[12]. Choi and Roh [18] propose a scheme to overcome the problem that eavesdroppers close to a tag can overhear messages sent from the tag. Yao *et al.* [29] present a random walk based approach for tradeoff between privacy and storage. Lim *et al.* [19]

TABLE II
Scanning costs(second) comparison in large-scale simulations

Tag size	AA	SMP	SEBA-2	SEBA-3
1000	72.95s	17.28s	15.18s	15.17s
2000	145.07s	21.30s	10.71s	10.70s
3000	213.46s	25.10s	9.192s	9.18s
4000	287.68s	29.23s	8.28s	8.27s
5000	362.26s	33.38s	7.69s	7.68s
6000	434.25 s	37.39s	7.29s	7.28s
7000	508.04s	41.50s	6.97s	6.95s
8000	579.6s	45.48s	6.72s	6.71s
9000	641.8s	48.97s	6.54s	6.53s
10000	712.48s	52.88s	6.37s	6.35s
11000	788.48s	57.12s	6.21s	6.19s
12000	866.57s	61.46s	6.07s	6.04s

present a randomized-bit-encoding scheme for strengthening the privacy protection for RFID tags.

Besides the deterministic identification or authentication, another important RFID scanning application is to use probabilistic techniques to retrieve some global features from large amount RFID tags, instead of identifying each single tag. [1], [17], [30]. Yang *et al.* [32] study the collision detection problem in RFID system. Tang *et al.* [31] study the capacity on RFID. Moreover, much more WSN approaches [33], [34] are also introduced to improve RFID efficiency in recent years.

VII. Conclusion

Existing RFID authentication methods require a pre-identification process, and suffer from high scanning cost and communication cost. We present the first identification-free batch authentication protocol, called Single Echo Batch Authentication (SEBA) for anti-counterfeiting. We conduct comprehensive analysis and trace driven simulations to evaluate this design. We believe that the techniques proposed in this paper will be great useful in RFID area for addressing anti-collision as well as privacy related issues. Our ongoing research will focus on designing more efficient outlier echo detections, and implementing our protocol on real RFID environment.

Acknowledgment

We would like to thank our shepherd, Prashant Krishnamurthy, and the anonymous reviewers for their valuable and helpful comments. We also thank Dong Xuan for his constructive suggestions. This work is supported in part by National Natural Science Foundation of China (NSFC) (No.60933003, No.60736016, No.60873262, and No.60903155), National High Technology Research and Development Program of China (863 Program) under Grants No.2009AA01Z116, National Basic Research Program of China (973 Program) under Grants No. 2011CB302705 and No. 2010CB328004, China Postdoctoral Science Foundation funded project (No.20090461298), Hong Kong Innovation and Technology Fund GHP/044/07LP and ITP/037/09LP, the Science and Technology Research and Development Program of Shaanxi Province under Grant No.2008KW-02, and IBM Joint Project.

References

[1] B. Sheng, C. C. Tan, Q. Li, and W. Mao, "Finding Popular Categories for RFID Tags," in Proceedings of ACM MobiHoc, 2008.

[2] T. Kriplean, E. Welbourne, N. Khoussainova, V. Rastogi, M. Balazinska, G. Borriello, T. Kohno, and D. Suciu, "Physical Access Control for Captured RFID Data," in *Pervasive Computing*, 2007.

[3] C. C. Tan, B. Sheng, and Q. Li, "Efficient Techniques for Monitoring Missing RFID Tags," in *IEEE Transactions on Wireless Communications*, 2010.

[4] L. M. Ni, Y. Liu, Y. C. Lau, and A. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID," in *ACM Wireless Networks*, 2004.

[5] Y. Liu, L. Chen, J. Pei, Q. Chen, Y. Zhao, "Mining Frequent Trajectory Patterns for Activity Monitoring Using Radio Frequency Tag Arrays," in Proceedings of IEEE PerCom, 2007.

[6] Y. Liu, Z. Yang, X. Wang, and L. Jian, "Location, Localization, and Localizability," in *Journal of Computer Science and Technology*, 2010.

[7] X. Zhang and B. King, "An Anti-Counterfeiting RFID Privacy Protection Protocol," in *Journal of Computer Science and Technology*, 2007.

[8] S. H. Choi and C. H. Poon, "An RFID-based Anti-counterfeiting System," *IAENG International Journal of Computer Science*, 2008.

[9] S. Weis, S. Sarma, R. Rivest, and D. Engels, "Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems," Lecture notes in Computer Science, 2004.

[10] T. Dimitriou, "A Secure and Efficient RFID Protocol that could make Big Brother (partially) Obsolete," in Proceedings of IEEE PerCom, 2006.

[11] L. Lu, J. Han, R. Xiao, and Y. Liu, "ACTION: Breaking the Privacy Barrier for RFID Systems," in Proceedings of IEEE INFOCOM, 2009.

[12] L. Lu, J. Han, L. Hu, Y. Liu, and L. M. Ni, "Dynamic Key-Updating: Privacy-Preserving Authentication for RFID Systems," in Proceedings of IEEE PerCom 2007.

[13] J. R. Cha and J. H. Kim, "Novel Anti-collision Algorithms for Fast Object Identification in RFID System," in Proceedings of ICPADS, 2005.

[14] "Information Technology Automatic Identification And Data Capture Techniques-Radio Frequency Identification For Item Management Air Interface. Part 6. Parameters for Air interface communications at 860-960 MHZ," ed: Standard ISO 18000-6, 2003.

[15] "EPCglobal Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz-960MHz," 2005.

[16] B. L. J and E. M, in *Introduction to probability and mathematical statistics*, 1995.

[17] M. Kodialam and T. Nandagopal, "Fast and Reliable Estimation Schemes in RFID Systems," in Proceedings of ACM MobiCom, 2006.

[18] W. Choi and B. H. Roh, "Backward Channel Protection Method for RFID Security Schemes Based On Tree-Walking Algorithms," in Proceedings of ICCSA, 2006.

[19] T. L. Lim, T. Li, and S. L. Yeo, "Randomized Bit Encoding for Stronger Backward Channel Protection in RFID Systems," in Proceedings of IEEE PerCom, 2008.

[20] W. Gu, X. Bai, S. Chellappan, D. Xuan and W. Jia, "Network Decoupling: A Methodology for Secure Communications in Wireless Sensor Networks," in *IEEE Transactions on Parallel and Distributed Systems*.

[21] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola, and V. Khandelwal, "Design and Implementation of PUF-Based Unclonable RFID ICs for Anti-Counterfeiting and Security Applications," in Proceedings of IEEE International Conference on RFID, 2008.

[22] S. R. Lee, S. D. Joo, and C. W. Lee, "An Enhanced Dynamic Framed Slotted ALOHA Algorithm For RFID Tag Identification," in Proceedings of IEEE MobiQuitous, 2005.

[23] B. Sheng, Q. Li, and W. Mao, "Efficient Continuous Scanning in RFID Systems," in Proceedings of IEEE INFOCOM, 2009.

[24] L. Xie, B. Sheng, C. C. Tan, H. Han, Q. Li, and D. Chen, "Efficient Tag Identification in Mobile RFID Systems," in Proceedings of IEEE INFOCOM, 2009.

[25] J. Myung and W. Lee, "Adaptive Binary Splitting: A RFID Tag Collision Arbitration Protocol for Tag Identification," in *Mobile Networks and Applications*, 2006.

[26] C. Law, K. Lee, and K. Y. Siu, "Efficient Memoryless Protocol For Tag Identification," in Proceedings of ACM DIALM Workshop, 2000.

[27] F. Zhou, C. Chen, D. Jin, C. Huang, and H. Min, "Evaluating and Optimizing Power Consumption of Anti-Collision Protocols for Applications in RFID Systems," in Proceedings of ACM ISLPED, 2004.

[28] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems," in Proceedings of SPC, 2003.

[29] Q. Yao, Y. Qi, J, Han, J, Xiangyang Li, and Y, Liu, "Randomizing RFID Private Authentication," in Proceedings of PerCom, 2009.

[30] C. Qian, H. Ngan, and Y. Liu, "Cardinality Estimation for Large-scale RFID Systems," in Proceedings of IEEE PerCom, 2008.

[31] S. Tang, J. Yuan, X. Li, G. Chen, Y. Liu, and J. Zhao, "RASPberry: A Stable Reader Activation Scheduling Protocol in Multi-Reader RFID Systems," in Proceedings of IEEE ICNP 2009.

[32] L. Yang, J. Han, Y. Qi, C. Wang, Y. Liu, Y. Cheng, and X. Zhong, "Revisiting Tag Collision Problem in RFID Systems," in Proceedings of IEEE ICPP, 2010.

[33] C. Wang, X. Li, C. Jiang, S. Tang, Y. Liu and J. Zhao, "Scaling Laws on Multicast Capacity of Large Scale Wireless Networks," in Proceedings of IEEE INFOCOM 2009.

[34] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, "A Spatiotemporal Protocol for Wireless Sensor Network," in *IEEE Transactions on Parallel and Distributed Systems*, 2005.