A Novel Coding Scheme for Secure Communications in Distributed RFID Systems

Kazuya Sakai, *Member, IEEE*, Min-Te Sun, *Member, IEEE*,
Wei-Shinn Ku, *Senior Member, IEEE*, and Ten H. Lai, *Senior Member, IEEE*

Abstract—Privacy protection is the primary concern when RFID applications are deployed in our daily lives. Due to the computational power constraints of passive tags, non-encryption-based singulation protocols have been recently developed, in which wireless jamming is used. However, the existing private tag access protocols without shared secrets rely on impractical physical layer assumptions, and thus they are difficult to deploy. To tackle this issue, we first redesign the architecture of RFID system by dividing an RF reader into two different devices, an RF activator and a trusted shield device (TSD). Then, we propose a novel coding scheme, namely Random Flipping Random Jamming (RFRJ), to protect tags' content. Unlike the past work, the proposed singulation protocol utilizes only the physical layer techniques that are already implemented. Analyses and simulation results validate our distributed architecture with the RFRJ coding scheme, which defends tags' privacy against various adversaries including the random guessing attack, correlation attack, ghost-and-leech attack, and eavesdropping.

Index Terms—RFID security, privacy, coding

1 Introduction

Radio frequency identification (RFID) technologies enable a tremendous amount of applications, such as supply chain management [1], electric transportation payment, and warehouse operations [2]. Objects and their owners are automatically identified by an attached RF tag, which causes the privacy threat to individuals and organizations. Thus, privacy protection is the primary concern when RFID applications are deployed in our daily lives. Since passive tags are computationally weak devices, encryption-based secure singulations [3] are not practical. Instead of relying on the traditional cryptographic operations, recent works [4], [5], [6] employ physical layer techniques i.e., jamming [7], to protect tags' data. With this approach, tags could be securely identified without pre-exchanged shared keys.

The issue with the existing solutions, the privacy masking [4], randomized bit encoding (RBE) [5], and dynamic bit encoding (DBE)/optimized DBE (ODBE) [6], is the impractical assumptions. In these solutions, all the bits transmitted by a tag are masked (jammed) under the assumption of an additive channel, where the receiver can read a bit only when 2 bits (the data bit and mask bit) are the same. When the 2 bits are different, it is assumed that the receiver is unable to recover the corrupted bit. However, this assumption is too strong since a reader should be able to detect signals from two different sources. In reality, a receiver of a data bit will decode it as either 0 or 1 without knowing the bit collision. If there is a bit collision, either the signal strength of data bits from the tag is stronger than that of the jamming bits, or vise versa. In other words, depending on the location of the reader, it can either read all the data bits or all the jamming bits. Also, masking requires the perfect synchronization between data bits and mask bits, which is difficult to achieve in practice.

In addition to this, DBE and ODBE have two drawbacks. One is encoding collision, where two different source data bits could be encoded into the same codeword. This causes the singulation process to fail. The other drawback is more serious. Tags' data encoded by DBE or ODBE could eventually be cracked, should an adversary repeatedly listen to the backward channel (i.e., signals from a tag to a reader). This approach is called *the correlation attack*. Moreover, none of the aforementioned solutions protect tags against ghost-and-leech attacks, i.e., impersonation of RF tags, similar to man-in-the-middle attacks.

To tackle these issues, we put forth a new RFID architecture and a novel coding scheme for privacy protection against various adversary models. The contributions of this paper are as follows:

- We redesign the system architecture of the non-encryption-based private tag access where an RF reader is divided into an RF activator and a TSD. The proposed architecture can be built by the current physical layer technologies, and thus our

- *K. Sakai is with the Department of Information and Communication Systems, Tokyo Metropolitan University, 6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan. E-mail: ksakai@tmu.ac.jp.*
- *M.-T. Sun is with the Department of Computer Science and Information Engineering, National Central University, Taoyuan 320, Taiwan. E-mail: msun@csie.ncu.edu.tw.*
- *W.-S. Ku is with the Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36849. E-mail: weishinn@auburn.edu.*
- *T.H. Lai is with the Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210. E-mail: lai@cse.ohio-state.edu.*

Manuscript received 5 June 2013; revised 10 Feb. 2014; accepted 9 Feb. 2015. Date of publication 15 Apr. 2015; date of current version 15 Jan. 2016. Recommended for acceptance by H. Jin.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2015.2423671

assumptions are much more practical than those of the existing solutions.
- The proposed distributed RFID architecture physically defends tags against ghost-and-leech attacks.
- We propose a novel coding scheme, named random flipping and random jamming (RFRJ), to protect the backward channel from passive adversaries, i.e., the random guessing attack, correlation attack, and eavesdropping. In our scheme, a tag/TSD randomly flips/jams a bit in a codeword and keeps the index of the these bits in secret. RFRJ guarantees that the TSD can recover a tag's content with one of the secrets, but an adversary cannot obtain the content of tags.
- Since the backward channel is protected by the RFRJ coding scheme, we can protect the forward channel (i.e., signals from a reader to a tag) by having an RF activator querying based on encoded data (or pseudo ID) space by RFRJ.
- We generalize the RFRJ coding scheme with the arbitrary source bits and codeword lengths. In addition, we prove the maximum information rate of our RFRJ scheme that achieves the perfect secret is 0.25.
- We conduct theoretical analyses for security of the proposed scheme, and prove that RFRJ provides perfect protection against passive attacks as long as jamming is successful.
- We evaluate our RFRJ coding scheme with the existing solutions by extensive simulations, and illustrate that the new architecture and coding scheme achieve our design goals.

The rest of this paper is organized as follows. Section 2 provides background knowledge for this research. We design a new RFID architecture in Section 3, and propose the RFRJ coding scheme in Section 4. Generalization of the RFRJ coding scheme is discussed in Section 5. Security analyses are provided in Section 6 and simulation results are demonstrated in Section 7. In Section 8, we review existing works for RFID security. Section 9 concludes this paper.

2 Preliminary

2.1 Physical Layer Security

Jamming is widely used for secure communications at the physical layer level, in which jamming signals corrupt receiving signals. Although this indicates that a legitimate receiver cannot decode received signals due to jamming, the full-duplex mode of wireless antennas allows the receiver to simultaneously transmit jamming signals and receive data. This can be done by canceling self-interference, in which transmitting signals interrupt receiving signals. According to [8], the current implementation can cancel self-interference up to 45 dB across 40 MHz. Therefore, with jamming techniques, an eavesdropper cannot steal communications unless it is in close proximity to a jamming source node.

It is known that perfect secrecy is possible without shared secrets by degrading the signal at an eavesdropper relative to that at the legitimate receiver [9]. Thus, jamming is a physical layer security technique suitable to wireless sensor networks where encryption-based security systems are not practical due to the power constraints of sensor nodes. Dialog code [7] is proposed that provides secure communications without shared secrets for wireless sensor networks. In this scheme, each source bit is encoded to a codeword, and jamming is performed during the transmission of the codeword. To achieve this, two assumptions must be held. One is that bit level jamming is possible; the other is that an eavesdropper cannot know which bit is jammed. Their implementation with sensor motes shows that both assumptions can be held by simulating a byte as a bit.

Another application of physical layer security with jamming is the protection of medical devices. In [10], a shield is developed to intermediate all the communications between a medical device of a patient and a reader from a doctor. A shield is capable of full-duplex communications, and protects the channel between a medical device and itself by jamming. Furthermore, the shield and the reader communicate with an encrypted channel. On detecting an unauthorized reader's access, the shield interrupts the communication by jamming all transmitted bits. The authors implemented the shield with a small portable device that looks like a necklace, and thus eavesdropping is almost impossible since an adversary must be at a very close position to the shield. By doing this, the proposed architecture does not need to modify medical devices in the markets.

2.2 Bit Level Jamming Models

Let b be a source bit, b_j be a jamming bit, and b' be the outcome of a bit b transmitted under jamming b_j. In [7], jamming channel models are categorized as follows.

- *Probabilistic flipping model*—no matter what value b_j has, the source bit b flips with the probability p_j, i.e., $P[b' \neq b] = p_j$.
- *AND channel model*—The receiver will decode $b' = 1$ when either b or b_j is 1. Otherwise, $b' = 0$.
- *XOR channel model*—The receiver will decode $b' = 1$ when $b \neq b_j$. Otherwise, $b' = 0$. It is known that one-time pad in this model can achieve perfect secrecy if the jamming bits are truly random in [11].
- *General model*—In this model, $P[b' = 0|b = 0, b_j = 0] + P[b' = 0|b = 0, b_j = 1] = 1$ and $P[b' = 0|b = 1, b_j = 0] + P[b' = 0|b = 1, b_j = 1] = 1$. The probability that $b' = 1$ is similar. This jamming model achieves perfect secrecy, since the probability that the receiver decodes $b' = 0$ is 0.5 whenever the jamming bits are truly random [7].

2.3 Distributed RFID Systems

In the traditional RFID system, an RF reader has two components, a transmitter (i.e., query transmission/energizing tags) and a listener (i.e., listening to a tag's reply) as shown in Fig. 1a, where a diamond represents the transmission function of a reader, a circle represents the listening function of a reader, and a rectangle represents a tag. The communication range of the backward channel is much shorter than that of the forward channel, and thus readers must be deployed based on the short-range backward channel to access all tags in the region as shown in Fig. 2a. A recent study proposes Distributed RF Sensing model [12] that employs two kinds of devices (a single RF transmitter and a

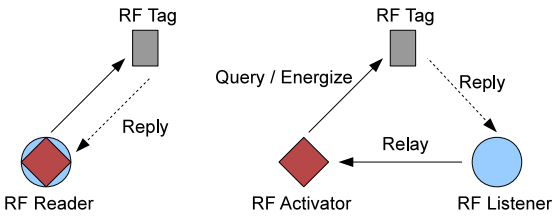


Fig. 1. Distributed RFID systems.

number of RF listeners) for each function of a reader as shown in Fig. 1b. The model contributes to cost reduction of RFID system deployment. For example, in Fig. 2, the traditional RFID system requires nine transmitters and nine listeners, while the distributed RFID system requires one transmitter and nine listeners.

3 Proposed Architecture

In this section, we propose a new RFID system architecture for a secure singulation as shown in Fig. 3.

3.1 Assumptions

We begin with listing physical layer assumptions as follows.

- Bit level jamming is feasible.
- An eavesdropper does not know if a bit is jammed.
- Probabilistic flipping model is used for a jamming environment.

As we discussed in Section 2, the first and second assumptions are already implemented and validated in [7], [13], [14]. On the other hand, there is no implementation of the backward channel protection methods in [4], [5], [6]. Therefore, our assumptions are much more practical than the past research.

3.2 New RFID System Architecture

Similar to [12], an RF reader is divided into two components, an *RF activator* and a *trusted shield device (TSD)*. In our new architecture, an RF activator queries a tag with a long-range signal (i.e., the forward channel) and energizes the tag. A TSD receives a tag's reply with a short-range signal (i.e., the backward channel), and it sends the reply to the activator via an encrypted channel, which we define as *the relay channel*. In typical RFID applications, a reader forwards tags' data to the back-end server. For simplicity, in this paper we consider the RF activator as the final destination of a tag's data by assuming the activator forwards collected data to the back-end server. A TSD works as an RF listener and it is

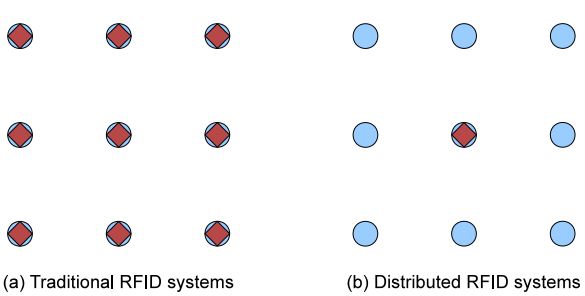


Fig. 2. Distributed RFID system deployment.

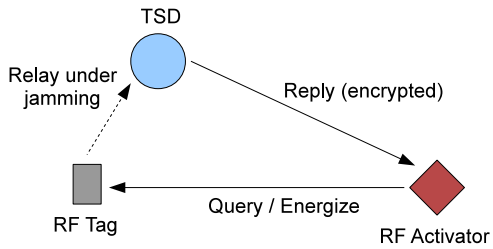


Fig. 3. The proposed RFID architecture.

capable of bit level jamming during reception of a tag's reply. Therefore, our new RFID system architecture consists of three components: an RF activator, a TSD, and RF tags.

In this paper, we introduce a new coding scheme, namely random flipping random jamming, for the backward channel protection. A tag will send encoded data (i.e., pseudo IDs) to a TSD under the jamming environment. This prevents adversaries from passive attacks, i.e., the random guessing attacks, correlation attacks, and eavesdropping. As we will show later, the RFRJ coding scheme ensures that adversaries cannot decode the original tag's ID from incomplete data due to jamming while the TSD successfully recovers the data from imperfect information.

A TSD is conceptually similar to the trusted masking device in [5] and a medical device shield implemented in [10], but different in the following functions.

- On overhearing a query from an activator to a tag, a TSD jams a bit in a codeword. As mentioned in the assumption, bit level jamming is possible.
- If an unauthorized reader tries to access a tag, a TSD jams against all bits of codewords so that the unauthorized reader cannot read the content of the transmitted data. A similar function is implemented in [10], where a shield device jams the whole communication on detecting unauthorized accesses. This can be done by letting an authorized activator communicate with a TSD before a singulation process.
- Unlike the trusted masking device and medical shield, a TSD intermediates only the backward channel.

With our new architecture, we can achieve the following design goals:

- The forward channel is protected by having an activator querying tag based on the pseudo ID space encoded by the RFRJ coding scheme.
- The RFRJ coding scheme protects the backward channel against the random guessing attacks, correlation attacks, and eavesdropping, as we will show in Section 6.
- Since we assume both an activator and a TSD have computational power, the relay channel can be protected by the traditional cryptographic operations.
- The proposed architecture defends against ghost-and-leech attacks. First, an adversary cannot forward an activator's query to a tag, since a TSD blocks all unauthorized accesses. Second, an adversary cannot obtain a tag's reply due to the jamming by TSD. Therefore, an adversary cannot impersonate a tag.

TABLE 1
Definition of Notations

Symbols	Definition
r	The RF Activator r
s	The TSD s
t	The RF tag t
b	The bit b
B	The source bits $\{b_1, b_2, ...\}$.
c	The codeword c
C	A domain of codewords $C = \{c_0, c_1, ...\}$
l_c	The length of a codeword $\|c\|$
l_b	The length of source bits $\|B\|$
I	The index of a bit in a codeword
$E(.)$	The function $E : \{0,1\}^{l_b} \rightarrow \{0,1\}^{l_c}$
$D(.)$	The function $D : \{0,1\}^{l_c} \rightarrow \{0,1\}^{l_b}$
$H(b, b')$	The Hamming distance between b and b'
$H(b, b', i)$	The Hamming distance between b and b' after removing the ith bit of b and b'
p_j	The probability that a jammed bit is flipped

- The physical layer assumptions are much more practical than the existing solutions [4], [5], [6], as we discussed in Section 3.1.

4 Random Flipping Random Jamming Coding

In this section, we present the random flipping random jamming coding scheme.

4.1 Definition

Let r be an RF activator, s be a TSD, and t be an RF tag. An activator which intends to obtain data from a tag sends a query on the forward channel. When the tag replies to the TSD, it encodes every l_b bits in the data into an l_c bits codeword with an encoding function $E(.)$. Note that l_b is not the length of an ID, but the unit to be encoded into a codeword. A coding scheme for private tag access is defined by the parameters, l_b, l_c, and C. Here, C is a set of codewords that could be used for encoding. During the transmission of a pseudo ID on the backward channel, the TSD conducts bit level jamming. On receiving the tag's reply, the TSD decodes the received codeword by a decoding function $D(.)$, and forwards the data to the activator via the relay channel.

In general, we call l_b-to-l_c the RFRJ coding scheme. For instance, the coding scheme with $l_b = 1$ and $l_c = 4$ is said to be the 1-to-4 RFRJ coding scheme. The notations utilized in this paper are listed in Table 1.

4.2 Private Tag Access Protocol

The proposed private tag access protocol works as follows. Suppose an RF activator r plans to read an RF tag t without disclosing the tag's ID to an eavesdropper. In this section, we first consider the length of the encoding unit l_b to be 1. Our idea can be applied to arbitrary values of l_b and l_c, where $l_b < l_c$. On receiving a request, the tag t extends a bit into an l_c-bit codeword, where $l_c \geq 4$ must hold. When the tag transmits data over the backward channel, it randomly selects a bit in a codeword and intentionally flips it. Note that this process is done before the tag sends out the codeword, so the data sent by the tag always contains a one-bit error. On the other hand, the TSD, which is an RF listener with jamming capability, jams a single bit in the codeword. The jamming causes the selected bit to flip. Let p_j $(0 \leq p_j \leq 1)$ be the probability that the bit jammed by the TSD is flipped. We denote I_s and I_t as the indexes of the selected bits by the TSD and the tag, respectively. The TSD randomly selects any bit in the first half of the l_c bits codeword, i.e., $1 \leq I_s \leq \lfloor \frac{1}{2} l_c \rfloor$, while a tag randomly selects a bit in the second half of the codeword, i.e., $\lfloor \frac{1}{2} l_c \rfloor + 1 \leq I_t \leq l_c$. By doing this, we can guarantee that the TSD and the tag do not select the same bit. Thus, the codeword received by the TSD or an eavesdropper contains a two-bit error when jamming flips the I_s-th bit and a one-bit error when jamming fails.

For instance, in Fig. 4, a source bit is encoded into a 4-bit codeword. The tag flips the third bit in the codeword, which is colored gray, and the TSD selects the first bit for jamming, which is crossed off.

Assume the original codeword is 1010. Since the tag flips the third bit, it will send 1000 over the backward channel. Meanwhile, the TSD jams the first bit. Hence, the TSD and the eavesdropper will receive $X000$, where X could be decoded to either 0 or 1. The TSD knows I_s, and thus it knows one of the three bits may contain an error after excluding the jammed bit. However, the eavesdropper does not know which bit the TSD jammed or which bit the tag flipped. For the eavesdropper, two out of the 4 bits may contain errors. Thus, the TSD and the eavesdropper have a different amount of information to decode the original codeword. In general, for 1-to-l_c, TSD knows that there is a 1-bit error out of $(l_c - 1)$ bits while the eavesdropper knows there is a two-bit error out of l_c bits at best.

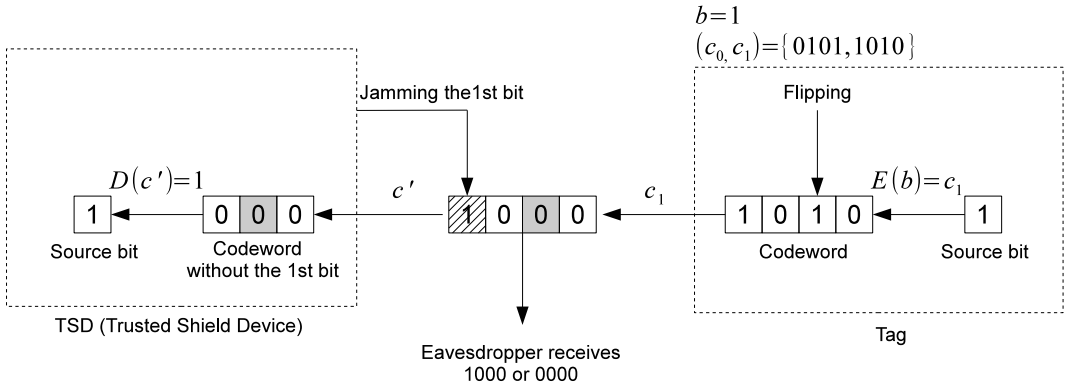


Fig. 4. The system model and basic idea.

Both the TSD and the tag keep the indexes of the bits they jammed/flipped in secret. The TSD has one of the secrets, but the eavesdropper knows neither of them. Therefore, with the coding scheme the receiver can decode a source bit when one of the $(l_c - 1)$ bits is flipped but not when two of the l_c bits are flipped. Our new system architecture and our proposed private access protocol allow for an RF activator to securely collect RF tags' content without shared secrets.

4.3 The Single Bit RFRJ Coding Scheme

We propose the RFRJ coding scheme with the parameter $l_b = 1$ and $l_c = 4$. Note that $l_c = 3$ does not work and $l_c = 4$ is the most efficient in terms of communication cost, which will be shown later. Let b be a source bit and c be a codeword. The encoding function $E : \{0,1\} \rightarrow \{0,1\}^4$ is defined by $E(b) = c_0$ if $b = 0$ and $E(b) = c_1$ if $b = 1$.

The encoding function $E(.)$ must ensure that the Hamming distance between c_0 and c_1, denoted by $H(c_0, c_1)$, is four. There are 16 such (c_0, c_1) pairs that can be used for private tag access. We call them *valid 4-bit codeword pairs*.

Definition 1 (Valid 4-bit codeword pairs). *When $l_c = 4$, a codeword pair (c_0, c_1), corresponding to a source bit pair $(0,1)$, is said to be valid when the Hamming distance between c_0 and c_1 is four, i.e.,*
$(0000, 1111)$, $(0001, 1110)$, $(0010, 1101)$, $(0100, 1011)$, $(1000, 0111)$, $(0011, 1100)$, $(0110, 1001)$, $(0101, 1010)$, *and* (c_1, c_0).

Let c' be the received codeword in which up to 2 bits could be flipped. We define the decoding function as $D : \{0,1\}^4 \rightarrow \{0,1\}$. Since a TSD knows the index of the jammed bit, the decoding function ignores the jammed bit. A tag also flips a bit which is unknown to the TSD, and the 3 bits contain the flipped bit after the TSD removes the jammed bit. Let $H(b, b', i)$ be the Hamming distance between b and b' after removing the ith bit from b and b'. $D(c')$ outputs 0 when $H(c', c_0, I_s) < H(c', c_1, I_s)$ and 1 when $H(c', c_0, I_s) > H(c', c_1, I_s)$. Note that $H(c', c_0, I_s) = H(c', c_1, I_s)$ never happens.

Next, we prove that the 1-to-4 RFRJ coding scheme successfully achieves our design goal.

Theorem 1. *When the RFRJ coding scheme with a valid codeword pair is used, the receiver can successfully decode the source bit, but the eavesdropper cannot.*

Proof. The TSD knows the value of I_s, so it can exclude the I_sth bit for the decoding process. Since a tag flips the I_tth bit where $I_s \neq I_t$, one of the 3 bits is flipped. Hence, this problem is reduced to whether or not the TSD can recover the original codeword sent by the tag, even if one out of three bits contains an error, while the eavesdropper cannot do it if two out of 4 bits contain errors. □

Let (c_0, c_1) be a codeword pair and c' be the codeword that the TSD and the eavesdropper receive. Since $H(c_0, c_1) = 4$, excluding the I_sth bit, $H(c', c_0, I_s)$ and $H(c', c_1, I_s)$ are both three. For instance, after removing the first bit of a codeword pair $(1100, 0011)$, we have $H(100, 011) = 3$. This implies that either c_0 or c_1 must be closer to c' than the other. Thus, the TSD can always decode it.

On the contrary, the eavesdropper does not know both I_s and I_t. All valid codeword pairs have the Hamming distance of four, and the 4-bit codeword received by the eavesdropper may contain a two-bit error. This indicates that $H(c_0, c') = H(c_1, c') = 2$, and the eavesdropper cannot decode it. Therefore, the claim is true.

Example. Consider a bit pair $(0,1)$ is mapped to one of a valid codeword pair, say $(c_0, c_1) = (0101, 1010)$, as shown in Fig. 4. A tag sends a bit 1 which will be encoded to 1010, and it selects the third bit to be flipped, i.e., $I_t = 3$. Afterward, the TSD selects the first bit for jamming, i.e., $I_s = 1$. Hence, the TSD will receive $X000$.

Let us mark the jammed bit by X. Since a tag flips a bit in the second half of the codeword, $X000$ contains a 1 bit error. With the 1 bit error in the second half of c_0 and c_1, we will have $c_0 = \{X100, X111\}$ and $c_1 = \{X000, X011\}$.

Clearly, sets of possible values of c_0 and c_1 are exclusive, and hence $H(X000, c_0, I_s) = H(X000, c_1, I_s)$ never happens. Thus, the TSD can always obtain the original codeword by taking the closer Hamming distance to $X000$. The decoding function takes c_1, and outputs 1.

On the contrary, the eavesdropper can neither derive the original codeword nor the source bit. When two of four bits have errors, i.e., 0000, the eavesdropper cannot distinguish whether the second and fourth bits of 0101 or the first and third bits of 1010 are flipped.

4.4 The 1-to-4 RFRJ Coding Scheme

We have illustrated how the RFRJ coding scheme encodes a single source bit to a 4-bit codeword. In general, an RF tag has data with arbitrary length or a constant length ID (e.g., 96-bit defined in EPC Class1 Gen2 [15]). In this section, we elaborate on the complete 1-to-4 RFRJ coding scheme.

In real RFID applications, a tag is likely to transmit the same data, such as its ID, to a TSD several times. Should an eavesdropper continuously listen, it can recover the content of the tag response by the help of the previous interrogations (the correlation attack [6]). To avoid the attack, we incorporate dependency by using different valid codeword pairs to each source bit.

Let b_k be the kth source bit that a tag intends to encode. To encode b_k, our coding scheme employs the previous source bits, b_{k-1}, b_{k-2}, b_{k-3}, and b_{k-4}. To be specific, we use the coding table in Table 2, where $b_k = 0$ if $k \leq 0$.

For example, the source bits with length four, 1010, will be encoded into four codewords with each having 4 bits, i.e., 1111 0011 1110 1001.

The decoding process is basically the same, but uses different codeword pairs for each source bit. The corresponding codeword for the b_kth source bit is obtained by Table 2. The decoding function $D(.)$ is applied to the received codeword c', computes $H(c', c_0, I_s)$ and $H(c', c_1, I_s)$, and then outputs 0 or 1.

The correctness of RFRJ is given by Lemma 2 and Theorem 3.

Lemma 2. *To successfully decode the kth source bit, a TSD must successfully decode the $(k-1)$th source bit.*

Proof. First, note that to decode the kth source bit, a TSD must know the previous source bits, b_{k-1}, b_{k-2}, b_{k-3}, and

TABLE 2
Coding Rule for the 1-to-4 RFRJ Coding Scheme

	$b_k = 0$	$b_k = 1$
$b_{k-4}b_{k-3}b_{k-2}b_{k-1}$	c	c'
0000	0000	1111
0001	0011	1100
0010	0001	1110
0011	1101	0010
0100	0101	1010
0101	1001	0110
0110	1000	0111
0111	1011	0100
1000	1111	0000
1001	1100	0011
1010	1110	0001
1011	0010	1101
1100	1010	0101
1101	0110	1001
1110	0111	1000
1111	0100	1011

b_{k-4}, which means that the receiver must have successfully decoded the $(k-1)$th source bit.

The proof is by contradiction. Assume that the TSD does not know b_{k-1} but knows all b_{k-2}, b_{k-3}, and b_{k-4}; then the TSD can decode b_k. Let $b_{k-4}b_{k-3}b_{k-2}b_{k-1}$ and $b'_{k-4}b'_{k-3}b'_{k-2}b'_{k-1}$ be two possible previous bit pairs, and the corresponding valid codeword pairs are $c = \{c_0, c_1\}$ and $c' = \{c'_0, c'_1\}$. By the assumption, $b_{k-4} = b'_{k-4}$, $b_{k-3} = b'_{k-3}$, $b_{k-2} = b'_{k-2}$, but $b_{k-1} \neq b'_{k-1}$. To decode b_k without decoding b_{k-1}, the Hamming distance between $H(c_0, c'_0)$, $H(c_0, c'_1)$, $H(c_1, c'_0)$, and $H(c_1, c'_1)$ must be more than two. However, all such codeword pairs have the Hamming distance two as shown in Table 2. This indicates the TSD cannot decode when one of two bits is flipped. Therefore, the TSD cannot decode b_k without decoding b_{k-1}, which leads to a contradiction. This concludes the proof. □

Example. Consider a TSD which successfully decodes $000X$ for $b_{k-4}b_{k-3}b_{k-2}$, but not b_{k-1}, where X could be 0 or 1. The two possible codeword pairs used to encode b_k are $(0000, 1111)$ and $(0011, 1100)$, and their corresponding source bits are 0000 and 0001, respectively. Clearly, $H(0000, 0011)$, $H(0000, 1100)$, $H(1111, 0011)$, and $H(1111, 1100)$ are all two. Thus, the TSD cannot decode the source bit b_k without decoding b_{k-1}.

Theorem 3. *A TSD can successfully decode all source bits encoded by the RFRJ coding scheme.*

Proof. The proof is by induction on k. □

Induction base. For the first source bit, the TSD knows the valid codeword pair since the base $b_{k-i} = 0$ $(1 \leq i \leq 4)$ as shown in Table 2. From Theorem 1, the TSD successfully decodes the first source bit.

Induction step. Assuming the TSD successfully decodes the kth source bit, we need to show it can decode the $(k+1)$th source bit. According to the RFRJ coding scheme, the TSD knows the previous bits for k to $k-4$ when it decodes the kth source bit. Thus, the TSD knows the valid codeword pair for the $(k+1)$th source bit from Table 2. From Theorem 1, the receiver successfully decodes the $(k+1)$th source bit. Therefore, the above claim is true.

Theorem 4. *When $l_b = 1$, the RFRJ coding scheme with $l_c = 4$ is the most efficient in terms of communication cost.*

Proof. We can prove the above claim by showing that the encoding with $l_c = 3$ does not work. A TSD will receive a 3-bit codeword where 1 bit is jammed and one bit is flipped. The proof is by contradiction. Assume the RFRJ encoding with $l_c = 3$ is the most efficient in terms of communication cost, then the TSD can decode the original codeword. If the TSD was able to decode the source bit, it would be able to recover the original codeword from the 2 bits where 1 bit is flipped after removing the I_sth bit from consideration. However, the Hamming distance between any pair of 2 bits is at most two, i.e., $H(00, 11)$, $H(11, 00)$, $H(01, 10)$, or $H(10, 01)$. Thus, when one of 2 bits is flipped, the TSD cannot recover the original codeword. This is a contradiction. The RFRJ coding scheme with $l_c = 3$ does not work. This completes the proof.

There are $8!$ coding tables that satisfy the property described in Lemma 2. Therefore, during initialization of an interrogation, an activator can send a query with the coding table number between $[1, 8!]$ to tags to prevent eavesdroppers from utilizing the disclosed bits from codewords in the previous interrogations. □

5 Generalization of RFRJ Coding

In this section, we consider general cases, the l_b-to-l_c coding scheme, where $1 \leq l_b < l_c$. Let E_{l_b,l_c} be an encoding function for l_b-to-l_c coding scheme which is defined by $E_{l_b,l_c} : \{0,1\}^{l_b} \rightarrow \{0,1\}^{l_c}$, and D_{l_b,l_c} be the corresponding decoding function. If 2 bits jamming and 2 bits flipping are considered, we can develop the $2-8$ coding scheme based on the 1-to-4 coding scheme. However, it is not interesting. Since the information rate is defined as $\frac{l_b}{l_c}$ $(0 < \frac{l_b}{l_c} \leq 1)$, the coding efficiency, in terms of the information rate of the 2-to-8 coding scheme, is the same as that of the 1-to-4 coding scheme. Therefore, the purpose of this section is to investigate the existence of any l_b-to-l_c coding scheme such that $\frac{l_b}{l_c} > \frac{1}{4}$.

First, we need to find valid codeword sets C that can be used for E_{l_b,l_c}. Note that we call C *codeword sets* instead of codeword pairs, since C contains more than two codewords when $l_b > 1$. In general, $|C| = 2^{l_b}$.

Intuitively, if every pair of codewords in C has the Hamming distance of four, C seems to be a valid codeword set. However, there is one restriction we need to enforce. Recall that a TSD jams the first half of a codeword and a tag flips the second half of the codeword to prevent the TSD and tag from selecting the same bit in the codeword. Considering this restriction, the following two properties are introduced to define a valid codeword set.

Property 1. *For a given codeword set C ($|C| = 2^{l_b}$), $\forall c, c' \in C$, $H(c_1, c'_1) \geq 2$ and $H(c_2, c'_2) \geq 2$, where $c = c_1||c_2$ and $c' = c'_1||c'_2$.*

Property 2. *For a given codeword set C ($|C| = 2^{l_b}$), $\forall c, c' \in C$, $H(c_1, c'_1) \leq 2$ and $H(c_2, c'_2) \leq 2$, where $c = c_1||c_2$ and $c' = c'_1||c'_2$.*

TABLE 3
Example of the 2-to-6 Coding

Source bits	Codewords
00	000000
01	110110
10	011011
11	101101

Property 1 is to ensure that a TSD can decode a codeword, and Property 2 is to prevent an eavesdropper from decoding under the RFRJ authentication. Note that any 4-bit codeword pairs have Properies 1 and 2. Now, we can define valid codeword sets, which can be used for the l_b-to-l_c coding scheme.

Definition 2 (Valid codeword sets). *Given l_b and l_c, a set of codewords with the properties 1 and 2 is said to be a valid codeword set.*

For example, the 2-to-6 coding scheme with Table 3 is valid. Consider the case that a tag replies a codeword 000000. Assume a TSD jams the first bit and the tag flips the fourth bit. If the jamming succeeds, an eavesdropper will receive 100100. All 000000, 110110, and 101101 with 2 bits flipping could be 100100, and thus the eavesdropper cannot decode the original codeword.

While the above 2-to-6 coding scheme is more efficient because its information rate is $\frac{1}{3}$, the perfect secrecy cannot be achieved. Since $l_b = 2$, there are four source bits, i.e., 00, 01, 10, and 11. Thus, the random guessing probability by an eavesdropper must be 0.25 for the perfect secret. In general, the random guessing probability must be $\frac{1}{2^{l_b}}$. However, in the aforementioned example, on receiving 100100 the eavesdropper can guess the original codeword to be either 000000, 110110, and 101101, but exclude the possibility of 011011. Hence, the eavesdropper narrows the source bits to be 00, 01, and 11. In other words, the correct guess probability is approximately 0.33.

When an eavesdropper receives a codeword, both the first and second half of the codeword contain one bit error. If the first and second half of any pair of codewords in a valid set has the Hamming distance of one, the original codeword could be any codeword in the valid set. This indicates that the eavesdropper cannot guess the original source bits with probability greater than $\frac{1}{2^{l_b}}$. To formally provide a valid codeword set for the perfect secrecy, we introduce Property 3.

Property 3. *For a given codeword set C ($|C| = 2^{l_b}$), let i ($1 \le i \le l_c$) be the index of a bit in a codeword $c \in C$. For $1 \le i \le \lfloor \frac{1}{2} l_c \rfloor$, $\forall c, c' \in C$, $H(c_1, c'_1, i) = 1$, and for $\lfloor \frac{1}{2} l_c \rfloor + 1 \le i \le l_c$, $H(c_2, c'_2, i) = 1$. Here, $c = c_1 || c_2$ and $c' = c'_1 || c'_2$.*

While the 1-to-4 coding scheme has Property 3, the aforementioned 2-to-6 coding scheme does not. In fact, there is no l_b-to-l_c coding scheme with $\frac{l_b}{l_c} > \frac{1}{4}$ that achieves the perfect secrecy. We prove this by Theorem 5.

Theorem 5. *The information rate of the l_b-to-l_c coding scheme that achieves the perfect secrecy is at most $\frac{1}{4}$.*

Proof. First, note that $l_b < l_c < 4l_b$ must hold for the information rate to be greater than $\frac{1}{4}$. The case when $l_b = 1$ is proven in Theorem 4. We will prove the case of $l_b > 1$ by showing there is no valid codeword set with Property 3 that achieves the perfect secrecy when $l_c < 4l_b$. The proof is by contradiction. Assume there exists such a valid codeword set C for some l_b and l_c with the information rate higher than $\frac{1}{4}$. Let us construct a graph with $2^{\lfloor \frac{l_c}{2} \rfloor}$ vertices, where each vertex has a $\lfloor \frac{l_c}{2} \rfloor$ bits length key and two vertices, say v_i and v_j, are connected if the Hamming distance of their keys is one. Any key has $\lfloor \frac{l_c}{2} \rfloor$ keys to which the Hamming distance is one, and so, as a result, each vertex has exactly $\lfloor \frac{l_c}{2} \rfloor$ neighboring vertices. Hence, each vertex is equivalent to each of the other vertices, and there are $2^{\lfloor \frac{l_c}{2} \rfloor}$ vertices in the graph. According to the definition of a hypercube [16], such a graph is a $\lfloor \frac{l_c}{2} \rfloor$-dimensional hypercube. The flipping of one bit in the key of a vertex will lead us to one of its neighboring vertices. Let v_i be a vertex whose key is the same as the first half (or second half) of a codeword. Let v_j be a neighboring vertex of v_i. Recall the first half (or second half) of a codeword received by an eavesdropper contains 1 bit error due to jamming (or flipping). To satisfy Property 3, the neighbor set of v_j must contain all vertices whose keys are the same as the first half (or second half) of codewords in C. Since each vertex has $\lfloor \frac{l_c}{2} \rfloor$ neighbors and there are 2^{l_b} codewords in C, $\lfloor \frac{l_c}{2} \rfloor \ge 2^{l_b}$ must hold. From the condition $l_c < 4l_b$, we can derive $2l_b - 1 \ge 2^{l_b}$, which never holds for any integer $l_b \ge 1$. Thus, there is no valid codeword set for the perfect secrecy if $l_b > 1$ and $l_c < 4l_b$. This is a contradiction. Therefore, the above claim must be true. □

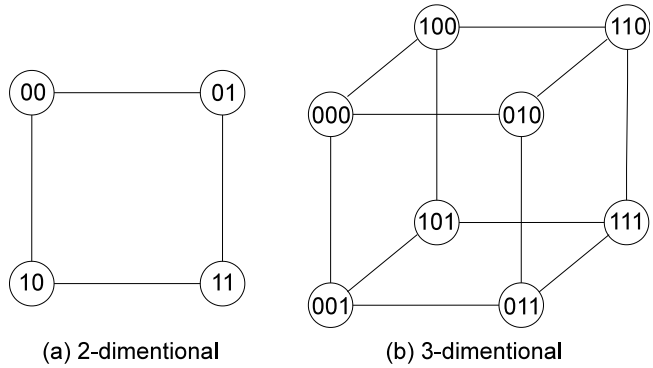


Fig. 5. The two and three-dimensional hypercube.

Example. For the 1-to-4 coding scheme, the source bit could be 0 or 1, and there are two codewords with length 4 in a valid codeword set. We can map the first half (or second half) of codewords to 2-dimensional hypercube ($\lfloor \frac{l_c}{2} \rfloor = 2$) as shown in Fig. 5a. The vertex pairs (00,11) and (10,01) satisfy Property 3. Thus, any combination of them can be a valid codeword pair which is listed in Definition 1 in Section 4.3. Consider the case $l_b = 2$. Theorem 5 indicates there is no coding scheme that achieves the perfect secrecy for $l_c < 2l_b = 8$. When $l_c = 6$ or $l_c = 7$, either the first or second half of a codeword is of length 3. Thus, we map the first half (or second half) of codewords to three-dimensional hypercube as shown in Fig. 5b. Assume the first half of a codeword is 000. After making 1 bit error

by jamming or flipping, the bit string refers to one of the neighbors. For instance, there is an error at the third bit, and we move to the vertex 001. An eavesdropper can guess the original first half of codeword is either 000, 011, or 101 (neighbors of the vertex 001) from the three-dimensional hypercube. Although there are four codewords, the eavesdropper can narrow the corresponding codeword to 3. Since each vertex in three-dimensional hypercube has only three neighbors, it is impossible to find a valid codeword set for $l_c < 8$. A similar argument holds for arbitrary l_b and l_c, where $l_b > 1$.

Therefore, the 1-to-4 coding scheme is the best in terms of the information rate and degree of security in our distributed RFRJ authentication. Note that we could develop i-to-$4i$ coding schemes with i bits jamming and i bits flipping by using the 1-to-4 coding scheme. However, such a discussion is trivial and less significant.

6 Security Analysis

In this section, we provide security analysis for the proposed coding scheme. Every source bit is assumed to be 0 or 1 with the same probability 0.5.

6.1 The 1-to-4 Coding Security

Let X be a random variable that represents the number of flipped bits in a codeword. The I_tth bit selected by a tag is always flipped with the probability 1, since this is done before the data is transmitted. On the other hand, the I_sth bit selected by a reader is flipped with the probability p_j, since the jamming does not guarantee that a target bit is flipped. In RFRJ, 1 or 2 bits in a codeword could be flipped depending on p_j. The probability that the events $X = 1$ and $X = 2$ occur is obtained by:

$$P[X = 1] = 1 - p_j, \tag{1}$$

$$P[X = 2] = p_j. \tag{2}$$

Since X is either 1 or 2, $P[X = 1] + P[X = 2] = 1$. In our 1-to-4 RFRJ coding scheme, an eavesdropper cannot decode when 2 bits are flipped. Thus, the eavesdropper cannot decode the source bit with the probability p_j. This rule is only applied to the first source bit, but not to the kth bit for $k > 1$ because it is encoded with a dependency.

Let X_k be a random variable that represents the number of flipped bits in the codeword corresponding to the kth source bit. Again X_k could be 1 or 2. Since a valid codeword pair used for the kth source bit is defined by the previous source bits, an eavesdropper must decode the $(k-1)$th source bit to successfully decode the kth source bit. Thus, the probability that the eavesdropper can decode the kth source bit is $P[X_k = 1|X_{k-1} = 1]$ with the base $P[X_0 = 1] = 1$. Although the selection of a valid codeword pair is dependent, $X_k = 1, 2$ and $X_{k-1} = 1, 2$ are independent events.

$$\begin{aligned} P[X_k = 1|X_{k-1} = 1] &= P[X = 1] \cdot P[X_{k-1} = 1] \\ &= P[X = 1]^k \\ &= (1 - p_j)^k. \end{aligned} \tag{3}$$

Hence, an eavesdropper has a very small chance to successfully decode the kth source bit when k is large.

6.2 Random Guessing Attacks

When the eavesdropper cannot decode, they may guess the source bit to be either 0 or 1 with even probability (i.e., the random guessing attacks). In this subsection, we consider the security of our coding scheme against an eavesdropper with random guessing capability. When a bit flipping by jamming fails, the eavesdropper decodes with the probability 1. Otherwise, it can successfully decode with the probability 0.5 by random guessing. Let b' be the bit decoded by the eavesdropper. Thus, the probability that the eavesdropper successfully decodes the source bit b is given by:

$$P[b = b'] = P[X = 1] + \frac{1}{2}P[X = 2]. \tag{4}$$

Let b_k and b'_k be the kth source bit and a bit decoded by the eavesdropper, respectively. We can obtain the probability that the random guessing succeeds at the kth source bit as follows:

$$\begin{aligned} P\big[b_k = b'_k\big] &= P\big[X_k = 1|b_{k-1} = b'_{k-1}\big] + \frac{1}{2}P\big[X_k = 2|b_{k-1} = b'_{k-1}\big] \\ &= \big(P[X = 1] + \frac{1}{2}P[X = 2]\big) \cdot P\big[b_{k-1} = b'_{k-1}\big] \\ &= \big(P[X{=}1] + \frac{1}{2}P[X{=}2]\big)^k \\ &= \left(1 - \frac{1}{2}p_j\right)^k. \end{aligned} \tag{5}$$

6.3 Anonymity Analysis

We will use the entropy based anonymity analysis that has been developed for coding schemes in [6]. Let n_b be the length of source bits (e.g., the data or tag ID length), and n_u be the number of source bits uncompromised by an eavesdropper. Then, the anonymity of the source bit is given by:

$$-\sum \frac{1}{2^{n_u}} log_2\left(\frac{1}{2^{n_u}}\right) \cdot \frac{1}{n_b} = \frac{n_u}{n_b}. \tag{6}$$

The average anonymity of our 1-to-4 RFRJ coding scheme is computed from the expected number of bits that an eavesdropper will decode. Let Z be the random variable that represents the number of compromised source bits. We will have $n_u = n_b - E[Z]$. From Equations (3) and (6), the average anonymity is computed by:

$$\frac{n_b - E[Z]}{n_b} = 1 - \frac{1}{n_b}\left(\sum_{k=1}^{n_b-1} kp_j(1 - p_j)^k + n_b(1 - p_j)^{n_b}\right). \tag{7}$$

Next, we formulate the anonymity at the ith interrogation cycle. Let Z_i be the number of compromised source bits at the ith interrogation cycle. $E[Z]$ in Equation (7) can be simplified by $\sum_{k=1}^{n_b} kp_j(1 - p_j)^k$. We can derive $E[Z_i]$ as follows:

$$E[Z_1] = \sum_{k=1}^{n_b} kp_j(1 - p_j)^k, \tag{8}$$

$$E[Z_2] = E[Z_1] + \sum_{k=1}^{n_b - E[Z_1]} kp_j(1 - p_j)^k, \tag{9}$$

$$E[Z_i] = E[Z_{i-1}] + \sum_{k=1}^{n_b - E[Z_{i-1}]} k p_j (1-p_j)^k \quad (10)$$

$$\approx i \sum_{k=1}^{n_b} k p_j (1-p_j)^k. \quad (11)$$

Therefore, the anonymity at the ith interrogation cycle is approximately $\frac{1 - iE[Z]}{n_b}$.

6.4 Analytical Results

According to [6], DBE and ODBE may generate the same pseudo ID from two different source IDs. Although such a possibility is very small, pseudo ID collisions cause the singulation process to fail. This is not acceptable. Contrarily, our RFRJ coding scheme does not have pseudo ID collisions by Lemma 6.

Lemma 6. *When $B \neq B'$ where B and B' are two sets of bits, $E(B) \neq E(B')$ always holds.*

Proof. The proof is by contradiction. Assume there exist two sets of bits, B and B', such that $E(B) = E(B')$, then there must exist $E(b_k)$ and $E(b'_k)$ where $b_k \neq b'_k$ and $b_{k-i} = b'_{k-i}$ for $1 \leq i \leq 4$. But, according to Table 2, this never occurs, since $B \neq B'$, there exists at least one bit pair $b_k \in B$ and $b'_k \in B'$ such that $b_k \neq b'_k$. This is a contradiction. Therefore, the claim must be true. □

DBE and ODBE have significantly improved the performance of the privacy masking [4] and RBE [5], especially against correlation attacks. Nevertheless, both DBE and ODBE cannot completely avoid correlation attacks. Hence, eventually the source bits are cracked. According to [6], encoded 96-bit data by ODBE with codeword length 4 and $p_j = 1$ is cracked in 800 interrogation cycles. However, our RFRJ is different. One of the important results in this paper is that the RFRJ coding scheme perfectly protects source bits from passive attacks when $p_j = 1$. This is proved by Theorem 7.

Theorem 7. *When $p_j = 1$, RFRJ achieves perfect secrecy against passive eavesdropping, random guessing, and correlation attacks.*

Proof. We will prove the above claim by showing that RFRJ coding results in the theoretical upper bound of anonymity and lower bound of random guessing probability. □

Eavesdropping—The probability that an eavesdropper can obtain source bits is given in Equation (3). When $p_j = 1$, Equation (3) results in $(1-p_j)^k = 0$. Thus, RFRJ provides perfect protection against passive eavesdropping.

Random guessing attacks—when all bits in a codeword are disclosed (jamming/masking fails for a codeword), an eavesdropper with the random guessing capability can decode the corresponding source bit with the probability 1. Otherwise, the source bit is successfully guessed with probability 0.5. Hence, the lower bound of the random guessing probability is 0.5^k for k-bit data. The random guessing probability for RFRJ is provided in Equation (5). When $p_j = 1$, we will have $(1 - \frac{1}{2}p_j)^k = 0.5^k$. This validates that RFRJ achieves the lower bound of the random guessing probability.

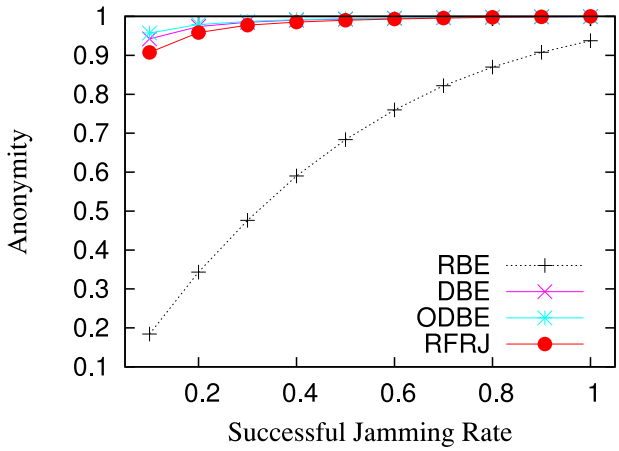


Fig. 6. Anonymity.

Correlation attacks—the upper bound of anonymity is 1. The anonymity of RFRJ for n_b bits source data is obtained by Equation (7). When $p_j = 1$, $P[X = 1] = 0$ and thus $E[Z] = 0$. Hence, the anonymity is 1. This holds for any $n_b \geq 1$, and encoded data by RFRJ is never cracked as long as $p_j = 1$. Thus, RFRJ avoids the correlation attacks.

Therefore, the claim is true.

7 Performance Evaluation

In this section, we demonstrate the performance of RFRJ with the existing secure coding schemes for RFID backward channels, including RBE [5], DBE, and ODBE [6].

7.1 Simulation Configurations

We have implemented the $1-\text{to}-4$ RFRJ coding scheme along with RBE, DBE, and ODBE. For fair comparisons, the codeword length for RBE, DBE, and ODBE is set to be four, which results in the same control overhead as the $1-\text{to}-4$ RFRJ coding scheme. In this simulation, data exchanged between an RF reader and RF tags are 96-bit tag IDs. Each tag encodes its ID with an encoding scheme and transmits it. Hundred RF tags are deployed in the reading range of an RF activator and TSDs. The reader executes a tree-based singulation protocol against encoded IDs. The successful jamming rate p_j varies from 0.1 to 1.0. For correlation attacks, a tag sends its ID under the RFRJ access protocol (or the privacy masking environment for RBE, DBE, and ODBE), and an eavesdropper keeps the scratches of disclosed data from previous interrogations. The number of interrogations for correlation attacks is set to be 1,000. For each configuration, 1,000 simulations were conducted.

7.2 Simulation Results

Fig. 6 shows the average anonymity of a pseudo ID by different encoding schemes with respect to the successful jamming rate p_j. All encoding schemes except RBE achieve very high anonymity. This implies that RFRJ has a strong protection against eavesdropping. In addition, we would like to emphasize that the physical layer assumptions used in our model are weaker than those in the privacy masking environment.

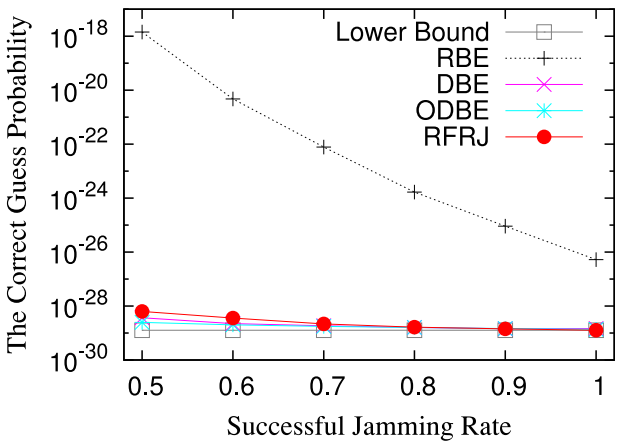


Fig. 7. Correct guess probability.

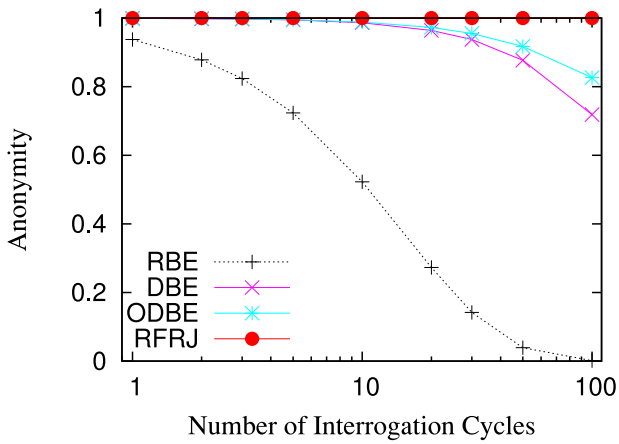


Fig. 9. Correlation attacks $p_j = 1.0$.

Fig. 7 illustrates the random guessing probability with respect to the successful jamming rate. Although RFRJ has a slightly higher random guessing probability than DBE and ODBE even when p_j is smaller than 0.7, it already provides a very strong protection. To be specific, when $p_j = 0.5$, the random guessing probability of RFRJ is 10^{-28}. It is clear that a random guessing eavesdropper has a very small probability of decoding the source bits.

Fig. 8 demonstrates the time required to crack all source bits by the correlation attacks with respect to the successful jamming rate. It is known that data encoded by RBE, DBE, or ODBE is eventually cracked due to design faults of the schemes. Contrarily, our RFRJ perfectly protects tags' IDs from the correlation attacks when $p_j = 1$. Note that the figure plots the results for p_j up to 0.95.

Figs. 9, 10, and 11 present the average anonymity of a pseudo ID by different encoding schemes with respect to the interrogation cycles for the successful jamming rate, 1.0, 0.9, and 0.8, respectively. For $p_j = 1.0$ (Fig. 9), RFRJ always has the maximum anonymity 1.0 because its design completely avoids the correlation attacks. This is one of the significant results of RFRJ. When $p_j = 0.9$, RFRJ achieves a similar anonymity to that of DBE and ODBE, and a much higher anonymity than that of RBE. When $p_j = 0.8$, RFRJ results in a slightly lower anonymity than that of DBE and ODBE. However, the difference is not significant.

7.3 Comparisons between Analytical and Simulation Results

In this subsection, the analytical and simulation results of the 1-to-4 coding scheme are compared to validate our analyses.

Fig. 12 shows the anonymity at the first interrogation cycle with respect to the jamming successful rate p_j. Fig. 13 illustrates the anonymity for different p_j with respect to the number of interrogation cycles. As can be seen in the figures, the analytical and simulation results are very close to each other. Thus, the simulation results validate our security analyses.

8 Related Work

8.1 RFID Security

In RFID systems, an RF reader must identify individual tags in its proximity by query and response. To effectively read a large number of tags, an anti-collision mechanism is critical to the performance of tag singulation protocols. In general, existing tag singulation protocols are classified into two categories, Aloha-based [15] and tree-walking-based [17]. Although these singulaiton protocols successfully identify every tag in a reader's vicinity, both of them do not provide privacy protection for the communication between readers and tags.

While it is desirable that the traditional symmetric and public/private key operations be used for private tag singulations, such an approach is not practical due to

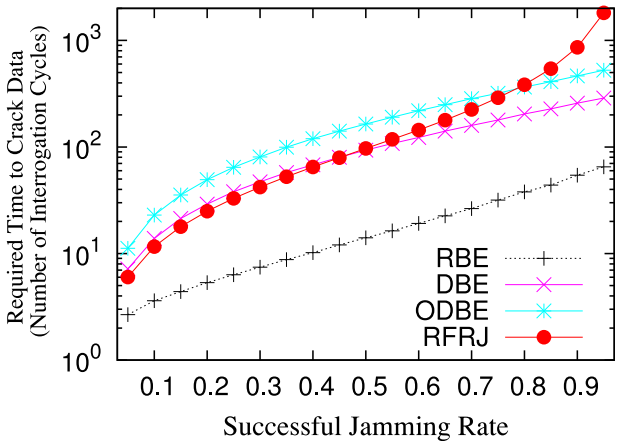


Fig. 8. Time to crack tag data.

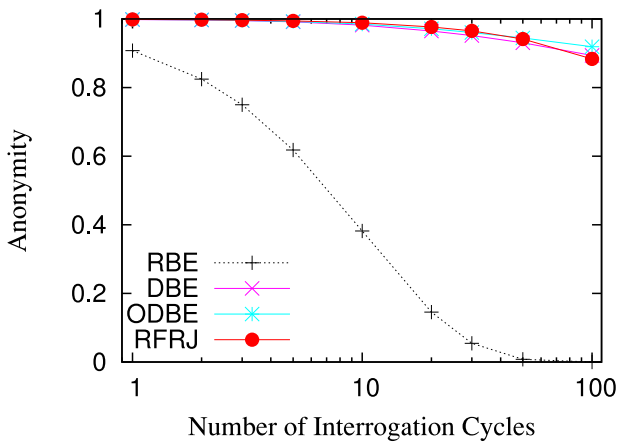


Fig. 10. Correlation attacks $p_j = 0.9$.

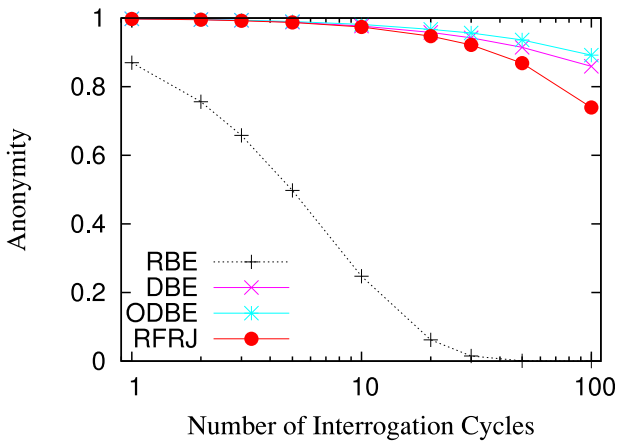


Fig. 11. Correlation attacks $p_j = 0.8$.

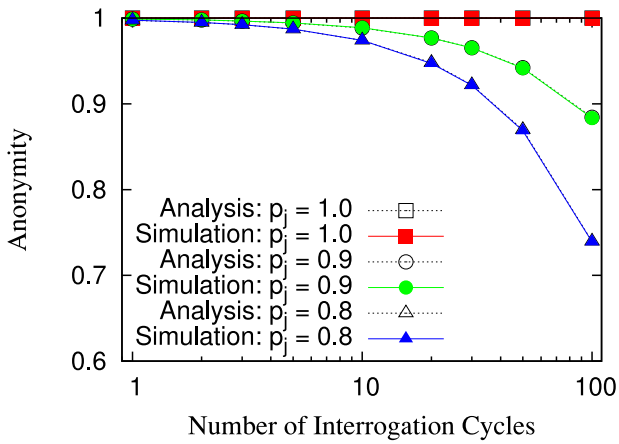


Fig. 13. Anonymity for correlation attacks.

computational power constraint of passive tags. This forces a number of encryption-based access protocols to use low-cost cryptographic operations [3], such as XOR, concatenations, hash functions, and so on. Although a reader successfully reads a tag without disclosing data to eavesdroppers, encryption-based singulation techniques require a large amount of overhead, including key exchanges/distributions [18] and structured key managements [19]. Therefore, in this paper, we focus on private tag authentication without a shared key.

8.2 Forward Channel Protection

In tree-walking-based protocols, each node is mapped to a leaf node of a binary tree comprised of the entire ID space, and a reader travels the tree in depth-first or breadth-first order by querying a prefix corresponding to an internal node in the tree. Thus, by eavesdropping the query, an adversary may obtain the tag's ID, and at least the tag's ID is partially disclosed. To protect the forward channel, the blinded tree-walking protocol [20] and the randomized tree-walking protocol [21] are proposed. In the blinded tree-walking protocol, instead of querying with a prefix that could be the entire ID in the worst case, a reader sends a next ID bit to avoid sending all bits in an ID. In the randomized tree-walking protocol, each tag maintains two IDs: a read tag ID and a pseudo ID generated by manufacturers or by the tag itself. A reader traverses the tree with the prefix of a pseudo ID and tags reply with their real ID. These techniques protect the forward channel, but not the backward channel.

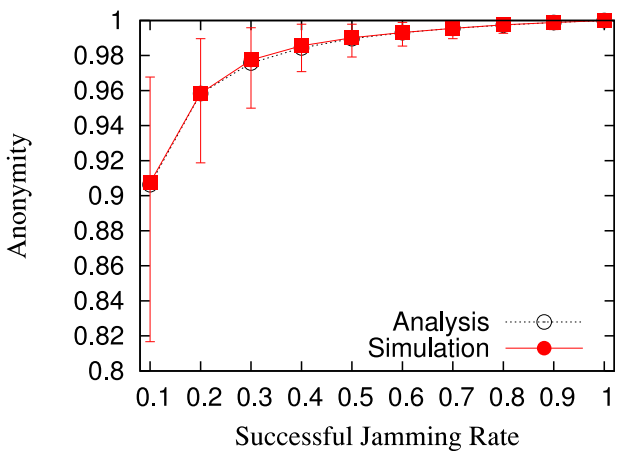


Fig. 12. Anonymity under different p_j values.

8.3 Backward Channel Protection

The most related studies to this paper are secure tree-walking-based singulations. Since tags can perform only simple functions, the protection of a tag's reply is much more difficult than the forward channel protection. To protect the backward channel without shared secrets, the physical layer security techniques are incorporated to the private tag access [4], [5], [6]. In privacy masking [4], a tag's reply is intentionally corrupted by mask bits (i.e., jamming under the additive channel). However, if the data sent by a tag and the mask bits are exactly the same, an adversary successfully eavesdrops the tag's content, called *the same bits problem*. RBE [5] alleviates the same bits problem by encoding by source bit to a codeword with a longer length. Nevertheless, RBE is vulnerable to the correlation attack, where an adversary listens to a tag's reply over several interrogations and recovers the source bits from scratches. To tackle this issue, DBE and ODBE [6] utilize the dependency among the source bits during their encoding process, and the information obtained in the previous interrogation is meaningless for the current interrogation. Note that RBE, DBE, and ODBE are used under privacy masking, and a reader composes a binary tree with pseudo IDs generated by these encoding schemes.

8.4 Ghost-and-Leech Attacks

Forward/backward channel protection techniques defend a tag's ID from passive adversaries, but not active adversaries. Ghost-and-leech attacks [22] are one of the active attacks in which an adversary impersonates a tag by forwarding a reader's query to the tag and the tag's reply to the reader. This attack is similar to the man-in-the-middle attacks in the study of cryptography. In [22], the author proposed Secret Handshake, where the user of a tag owner defines a motion signature, e.g., motion of a circle, a triangle, an alpha, etc., and unlocks the tag before a reader accesses it. However, this solution only works for the applications in which a tag is used for the owner's identification, such as ID cards, since the motion signature

must be defined for individual tags. Hence, this approach cannot be applied to RFID systems where tags are attached to products, e.g., supermarkets, library, supply chains, and more.

9 Conclusion

RFID systems serve as an enabling technology for the Internet of Things. However, security concerns of existing RFID systems have become a major obstacle for their wide adoption. The RFID protection mechanisms in the literature either work for only a few specific attacks or have unrealistic physical layer assumptions. In this paper, we first propose a novel distributed RFID architecture which divides the RF reader into two parts: an RF activator and a TSD, each tailoring for a specific function of an RF reader. In addition, we propose the RFRJ coding scheme, which when incorporated with the new architecture, works against a wide range of adversaries including the random guessing attack, correlation attack, ghost-and-leech attack, and eavesdropping. The physical layer assumptions of the proposed RFID architecture and the encoding scheme are readily available. In addition, the hardware cost of the new architecture is theoretically cheaper than the existing RFID systems. We believe the proposed architecture will serve as the foundation of the next-generation RFID systems.

Acknowledgments

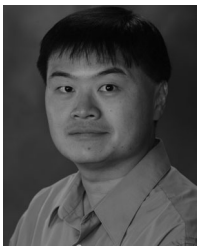
Min-Te Sun is the corresponding author.

References

[1] Y. Li and X. Ding, "Protecting RFID communications in supply chains," in *Proc. 2nd ACM Symp. Inf., Comput. Commun. Security*, 2007, pp. 234–241.
[2] H. K. H. Chow, K. L. Choy, W. B. Lee, and K. C. Lau, "Design of a RFID case-based resource management system for warehouse operations," *Expert Syst. Appl.*, vol. 30, no. 4, pp. 561–576, Feb. 2006.
[3] A. Juels,"RFID security and privacy: A research survey," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 381–394, 2006.
[4] W. Choi, M. Yoon, and B.-h. Roh, "Backward channel protection based on randomized tree-walking algorithm and its analysis for securing RFID tag information and privacy," *IEICE Trans.*, vol. 91-B, no. 1, pp. 172–182, 2008.
[5] T.-L. Lim, T. Li, and S.-L. Yeo, "Randomized bit encoding for stronger backward channel protection in RFID systems," in *Proc. IEEE 6th Annu. Int. Conf. Pervasive Comput. Commun.*, 2008, pp. 40–49.
[6] K. Sakai, W.-S. Ku, R. Zimmermann, and M.-T. Sun, "Dynamic bit encoding for privacy protection against correlation attacks in RFID backward channel," *IEEE Trans. Comput.*, vol. 62, no. 1, pp. 112–123, Jan. 2013.
[7] L. Sang, "Designing physical primitives for secure communication in wireless sensor networks," Ph.D. dissertation, Department of Computer Science and Engineering, The Ohio State University, 2010.
[8] M. Jain, J. L. Choi, T. M. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha, "Practical, real-time, full duplex wireless," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 301–312.
[9] A. D. Wyner,"The wire-tap channel," *Bell Syst Tech. J.*, vol. 54, no. 8, pp. 1355–1387, 1975.
[10] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They can hear your heartbeats: Non-invasive security for implantable medical devices," in *Proc. ACM SIGCOMM Conf.*, 2011, pp. 2–13.
[11] H. Delfs and H. Knebl, *Introduction to Cryptography: Principles and applications*, 2nd ed. New York, NY, USA: Springer, 2007.
[12] D. D. Donno, F. Ricciato, L. Catarinucci, A. Coluccia, and L. Tarricone, "Challenge: Towards distributed RFID sensing with software-defined radio," in *Proc. 16th Annu. Int. Conf. Mobile Comput. Netw.*, 2010, pp. 97–104.
[13] L. Sang and A. Arora, "A shared-secret free security infrastructure for wireless networks," *ACM Trans. Auton. Adaptive Syst.*, vol. 7, no. 2, pp. 23:1–23:21, 2012.
[14] L. Sang and A. Arora, "Capabilities of low-power wireless jammers," in *Proc. INFOCOM*, 2009, pp. 2551–2555.
[15] EPCglobal, EPC radio-frequency identity protocols Class-1 Generation-2 UHF RFID Protocol for communications at 860 MHz-960MHz version 1.0.9 [Online]. Available: http:www.epcglobalinc.org/standards, 2005.
[16] J. B. Wilker, "An extremum problem for hypercubes," *J. Geometry*, vol. 55, pp. 174–181, 1996.
[17] J. Myung, W. Lee, J. Srivastava, and T. K. Shih, "Tag-splitting: Adaptive collision arbitration protocols for RFID tag identification," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 6, pp. 763–775, Jun. 2007.
[18] A. Juels, R. Pappu, and B. Parno, "Unidirectional key distribution across time and space with applications to RFID Security," in *Proc. USENIX Secur. Symp.*, 2008, pp. 75–90.
[19] M. E. Hoque, F. Rahman, and S. I. Ahamed, "AnonPri: An efficient anonymous private authentication protocol," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2011, pp. 102–110.
[20] 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 *Proc. 1st Int. Conf. Security Pervasive Comput.*, 2003, pp. 201–212.
[21] S. A. Weis, "Security and privacy in radio-frequency identification devices," Masters Thesis, Department of Electrical Engineering and Computer Science, MIT, 2005.
[22] A. Czeskis, K. Koscher, J. R. Smith, and T. Kohno, "RFIDs and secret handshakes: Defending against ghost-and-leech attacks and unauthorized reads with context-aware communications," in *Proc. 15th ACM Conf. Comput. Commun. Security*, 2008, pp. 479–490.

Kazuya Sakai (S'09-M'14) received the PhD degree in computer science and engineering from The Ohio State University in 2013. Since 2014, he has been an assistant professor in the Department of Information and Communication Systems, Tokyo Metropolitan University. His research interests are in the area of wireless networks, mobile computing, and network security. He is a member of the IEEE.

Min-Te Sun (S'99-M'02) received the BS degree in mathematics from National Taiwan University in 1991, the MS degree in computer science from Indiana University in 1995, and the PhD degree in computer and information science from The Ohio State University in 2002. Since 2008, he has been with the Department of Computer Science and Information Engineering, National Central University, Taiwan. His research interests include distributed algorithm design and wireless network protocol development. He is a member of the IEEE.



Wei-Shinn Ku (S'02-M'07-SM'12) received both the MS degrees in computer science and in electrical engineering in 2003 and 2006, respectively, and the PhD degree in computer science all from the University of Southern California (USC) in 2007. He is an associate professor with the Department of Computer Science and Software Engineering, Auburn University. His research interests include spatial data management, mobile data management, geographic information systems, and security and privacy. He has published more than 70 research papers in refereed international journals and conference proceedings. He is a senior member of the IEEE.

Ten H. Lai is a professor of computer science and engineering at the Ohio State University. He is interested in applying Zen to teaching and research. He served as program chair of ICPP 1998, a general chair of ICPP 2000, a program co-chair of ICDCS 2004, a general chair of ICDCS 2005, and recently, a general co-chair of ICPP 2007. He is/was an editor of the *IEEE Transactions on Parallel and Distributed Systems*, *ACM/Springer Wireless Networks*, *Academia Sinica's Journal of Information Science and Engineering*, *International Journal of Sensor Networks*, and *International Journal of Ad Hoc and Ubiquitous Computing*.