Novel Trusted Hierarchy Construction for RFID Sensor–Based MANETs Using ECCs

Adarsh Kumar, Krishna Gopal, and Alok Aggarwal

In resource-constrained, low-cost, radio-frequency identification (RFID) sensor–based mobile ad hoc networks (MANETs), ensuring security without performance degradation is a major challenge. This paper introduces a novel combination of steps in lightweight protocol integration to provide a secure network for RFID sensor–based MANETs using error-correcting codes (ECCs). The proposed scheme chooses a quasi-cyclic ECC. Key pairs are generated using the ECC for establishing a secure message communication. Probability analysis shows that code-based identification; key generation; and authentication and trust management schemes protect the network from Sybil, eclipse, and de-synchronization attacks. A lightweight model for the proposed sequence of steps is designed and analyzed using an Alloy analyzer. Results show that selection processes with ten nodes and five subgroup controllers identify attacks in only a few milliseconds. Margrave policy analysis shows that there is no conflict among the roles of network members.

Keywords: Access control policies, analysis and modeling, lightweight protocols, MANETs, readers, RFID, tags.

Manuscript received Feb. 10, 2014; revised June 16, 2014; accepted July 10, 2014.

Adarsh Kumar (corresponding author, adarsh.kumar@jiit.ac.in), Krishna Gopal (mesoft_adarsh@rediffmail.com), and Alok Aggarwal (director@jpiet.com) are with the Department of Computer Science Engineering and Information Technology, Jaypee Institute of Information Technology, Uttar Pardesh, India.

I. Introduction

Radio-frequency identification (RFID) devices, with the help of tags and reader devices, are used for object identification and any subsequent tracking and record management. Wireless sensor nodes have sensing, ad hoc, wireless, and multi-hops communication capabilities. Security in RFID sensor–based mobile ad hoc networks (MANETs) demands that we discuss the following issues:

- Areas: routing, data forwarding, link layer, and so on.
- Attacks: external versus internal, passive versus active, and so on.
- Defense mechanisms: confidentiality, integrity, and so on.
- Challenges: lightweight, decentralization, fault tolerant, and so on.
- Protocols: symmetric key cryptography, digital signature, and so on.
- Applications: disaster recovery, public safety, and so on.

Various RFID and sensor-node integration mechanisms have been proposed [1]–[3]. These scenarios help to construct MANETs when RFID devices are attached with mobile sensor devices [4]. The mobile sensor devices are then able to sense nearby objects from which to collect necessary information. For example, a MANET may consist of RFID readers attached with sensors to help route information and construct ad hoc networks, or it may consist of tags attached with sensor nodes that communicate among themselves to construct ad hoc networks, and so on [2]. Such integration and construction has many advantages. For example, it increases the range of availability of information for animal, person, or object counting; record management of large datasets; observations of insurgence activities during military exercises; sensing of human body activities in healthcare environments; and so on.

http://dx.doi.org/10.4218/etrij.15.0114.0177

However, this integration carries with it many network security threats, such as jamming, spoofing, tracking, flooding, physical attacks, and so on. Both RFID and sensor devices are resource-constrained devices; thus, they require lightweight or ultra-lightweight cryptography.

This paper extends the virtual node–based hierarchy construction found in [5] and [6]. First, an extension of the creation of multiple hierarchies is made, in which each subgroup consists of a fixed number of mobile sensor nodes. The nodes are then divided into subgroups, hierarchies, and networks to reduce the funneling effect [7]. Identification marks to these nodes are assigned using compact ECCs. Second, under the extension, the locations of the mobile sensor nodes are observed for the establishment and maintenance of the relationship policies using trust management. These policies are analyzed and verified against conflicts of interest using a Margrave policy analysis [8]–[9]. A lightweight model of the proposed scheme is constructed and analyzed using an Alloy analyzer [10]–[12].

The rest of this paper is organized as follows. Related work to lightweight cryptography, identification protocols using ECC, authentication protocols, and grouping protocols with distance bounding, as well as a discussion on the possibilities of attacks, is given in Section II. Section III states the definitions, symbols, and notations used in this work. A novel identification, authentication, group-distance bounding, ownership transfer mechanism–based integrated approach is presented in Section IV. In Section V, the proposed mechanism is checked through probability-based fault acceptance rate analysis for Sybil, eclipse, and desynchronization attacks. Section VI shows the lightweight modeling, primitives, and policy analysis. Finally, conclusions are drawn in Section VII.

II. Background

1. Lightweight Cryptography

Lightweight cryptography can be classified into primitives and protocols. The goal of lightweight primitives is to achieve confidentiality, integrity, authentication, availability, and non-repudiation [6]. Lightweight primitives can be classified as being either symmetric or asymmetric. Lightweight protocols can be classified into the following: identification, authentication, distance bounding, grouping proof, and tag ownership transfer. After assigning identifications to nodes, communication between nodes occurs over an insecure channel; thus, there is a need for strong mechanisms to secretly exchange information over the channel. In such cases, authentication protocols are designed to fill this need. Network services are administrated through local and global subgroup constructions so as to save energy consumption. This goal is achieved through distance-bounding and group-yoking protocols. The process of joining and leaving nodes for right management, tag transfer, or updating some secret information is achieved through tag and ownership transfer protocols. The development of these protocols over recent years is discussed in the next subsection.

2. Identification Protocols Using ECCs

In code-based identification systems, the McEliece-based cryptosystem is the first public-key linear ECC–based cryptosystem using Goppa codes, and it is NP complete [13]. This scheme has fast encryption and decryption functions and is efficient for resource-constrained embedded devices [14]. The major problem of this scheme is its large key size [15]. Various schemes have been proposed to reduce the key size using different coding techniques. The Sidelnikov system with Reed-Muller codes is an example of a reduced key length code–based cryptosystem. However, this system is prone to structural attacks [16]. The Janwa–Moreno system with algebraic geometric codes was broken by Faure and Minder for codes on curves of genus $g \leq 2$ [17]. The Gabidulin–Paramonov–Tretjakov system with Gabidulin codes also proposed a reduced key length cryptosystem. The aforementioned systems are affected by trap door and structural attacks, such as Gibson's attack, Ourivski's attack, and polynomial time attacks [18]–[19]. The Niederreiter system with generalized Reed–Solomon codes is also designed for identification. But, this system does not work properly with matrix arithmetic operations [20]. Berger and others proposed a reduced key generation process using Reed–Solomon codes in quasi-cyclic form [15]. This coding mechanism is selected in this paper to generate random codes and to assign identification marks for the current work.

3. Authentication Protocols

Authentication through location verification was initially proposed by Sastry and others [21]. This time-based location verification approach requires extra hardware to verify locations. Capkun and others also proposed an on-the-spot verification algorithm that again requires extra hardware [22]. To remove the cost of such extra hardware, various lightweight techniques have been proposed [23]. Techniques that do not require extra hardware are classified as: trust score–based techniques [24], distance-based techniques [25], or probabilistic approaches [26]. These techniques are affected with basic logic malfunctioning attacks [27]. In 2012, Tian and others proposed a permutation-based authentication protocol to avoid attacks due to unbalanced OR and AND operations by

using permutations [26]. Permutation-based protocols are prone to desynchronization, traceability, and disclosure attacks. However, attempts have been made to modify RFID authentication protocols with permutation protocols to avoid desynchronization attacks [27].

4. Grouping Protocols

In 2004, Juels was the first to propose the idea of generating evidence for the presence of a group of nodes [28]. Juels' idea of grouping is for un-trusted users and was proposed to provide verifiable proofs [28]–[29]. Lamport extended this idea into two mechanisms: basic construction and one based on Lamport digital signature construction [29]. However, this idea of grouping is prone to replay attacks, denial of service, and impersonation attacks. Saito and Sakurai, in 2005, proposed a timestamp-based grouping scheme [30]. This idea was to remove Juels' replay attack. However, Piramuthu found that the extended approach was, indeed, not protected from replay attacks [31]. In 2007, Cho and others proposed a new mechanism to protect a group of nodes from replay attacks [32]. In their scheme, increased complexity made it difficult for an attacker to instigate a replay attack. In 2010, Lee and others proposed an ECC-based yoking protocol [33]. In this scheme, computations on tags are simpler and more efficient than in other schemes. Recently, another lightweight authentication protocol that takes into consideration anonymity, authentication, and untraceability was designed by Chien [34]. This system is based on ECCs and a public key–based Rabin cryptosystem.

5. Attacks

Dynamic topology, an open medium, distributed nodes, fluctuating link capacity, limited energy, scarcity of resources, and so on make a system vulnerable to attacks. External and internal attackers impersonate the identities and privileges of other nodes so as to disturb a network. Sybil, eclipse, and desynchronization are examples of such control traffic attacks and are described as follows:

- *Sybil attack*. In this type of attack, a malicious node behaves as a legitimate node by impersonating or stealing the identity of a node. This attack is possible through methods such as direct communication, indirect communication, fabricated or stolen identities, and so on. This attack cannot be prevented completely until the attacker has enough resources to create the Sybils. In the wider literature, various methods to limit this type of attack have been proposed. These methods can be classified based on the following parameters:
  - Strong identification of peers [35]
  - Authentication and authorization [36]
  - Trust score [37]
- *Eclipse attack*. In a network, many versions of this attack are possible. This attack does not directly disrupt the network, but it does boost other attacks. In this attack, an attacker tries to capture neighboring nodes that can be easily compromised. Malicious nodes conspire to mislead correct nodes to control traffic. This enables flooding, denial of service, or censorship attacks. Mechanisms to defend against this kind of attack are classified into the following two major categories [38]–[39]:
  - Structural constraints
  - Proximity constraints
- *Desynchronization attack*. An attacker can break the synchronization between reader and tag to use different identity pseudonyms. Different identity pseudonyms between reader and tag result in a desynchronization attack [40]. In [41], a desynchronization attack is analyzed over a hash-based mutual authentication scheme. This work states that a desynchronization attack can be avoided by adding additional messages to acknowledge the identity pseudonyms. However, Deursen and Radomirovic analyzed that sending an acknowledge packet does not prevent such an attack [42].

III. Preliminaries

1. Definitions

Definition 1 (Lightweight memory-less sensor source). Let us denote a given mobile sensor node by MN. Let MN follow a particular path $SP = s_1, s_2, \ldots, s_p$, where s_a is an independent random state such that $1 \leq a \leq p$ for $p \in \mathbb{Z}^+$. Then, the probability that MN follows a particular state s_a is given by $P(\text{MN follows } s_a) = \text{MN}_{s_a}$. This probability is independent of previous or future states, but a hidden Markov chain can be used to relate the states and their respective probabilities. It is assumed to be memory-less because the path followed is independent of any stored route. Further, it is based on key distribution, authentication, availability, or any other cryptographic primitive.

2. Symbols and Notation

Let $\text{MN}_{(i,j)}^{(k,n)}$ represent the ith mobile sensor node (MN) of the jth subgroup (SG) in the kth hierarchical layer (HL) of the nth network (NW). Here, $i \in \{1, 2, \ldots, W_{\text{MN}}\}$, $j \in \{1, 2, \ldots, X_{\text{SG}}\}$, $k \in \{1, 2, \ldots, Y_{\text{HL}}\}$, and $n \in \{1, 2, \ldots, Z_{\text{NW}}\}$, where W_{MN} is the maximum number of mobile nodes in each subgroup, X_{SG} is the maximum number of subgroups among all hierarchical layers, Y_{HL} is the highest hierarchical layer, and Z_{NW} is the maximum number of networks constructed. Let F_q denote a finite field (F) with primitive element G and prime q.

Suppose $E_K(m)$ and $D_K(m)$ are the encryption and decryption of message m using key K, respectively. Let RSSI, A_{TS}, and $Q(\mathrm{MN}_{(i,j)}^{(k,n)})$ represent the reserved signal strength indicator (RSSI), the average trust score of an application or mobile node, and entropy of $\mathrm{MN}_{(i,j)}^{(k,n)}$, respectively.

IV. Proposed Methodology

Nodes in the proposed work are divided into subgroups, groups, and hierarchies. Figure 1 shows the construction of a single hierarchy. Each subgroup consists of a fixed number of mobile nodes. These mobile nodes may be in the form of real nodes or virtual programmable nodes. Real nodes are existing nodes, and virtual nodes are programmable nodes; this allows us to construct subgroups that are in the same vicinity as each other to save energy [5].

An increase in W_{MN}, X_{SG}, Y_{HL}, or Z_{NW} will lead to an increase in the traffic intensity, network overhead, delays, collisions, congestion, packet loss, and power consumption. This would then result in a decrease in throughput [7]. We now outline the steps necessary to construct a secure hierarchy.

1. Unique Number Generation Using Compact Codes

Berger and others used the process of key generation using Reed–Solomon codes in quasi-cyclic form to generate identification numbers with reduced key length [15]. This process is extended in this work to generate keys for distributing identification marks using a lightweight code–based identification protocol. In F_q, a tuple [TU = 1, T, … , T^{g-1}] is selected, where $T = (G)^{(q^I-1)\times g}$. Here, g is the order of T, and I is a random number belonging to the set {0, 1, … , q}. If d is the hamming distance in binary codes and $\boldsymbol{V}$ is the block parity-check matrix, then a Reed–Solomon code in quasi-cyclic form can be represented by $M_{2I} = (\boldsymbol{V}_0 \,|\, \boldsymbol{V}_1 \cdots | \boldsymbol{V}_{(q^I-1)-1})$, and $\boldsymbol{V}_d$ can be calculated as

$$\boldsymbol{V}_d = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ G^d & G^dT & \cdots & G^dT^{g-1} \\ (G^d)^{2I-1} & (G^dT)^{2I-1} & \cdots & (G^dT^{g-1})^{2I-1} \end{pmatrix}.$$

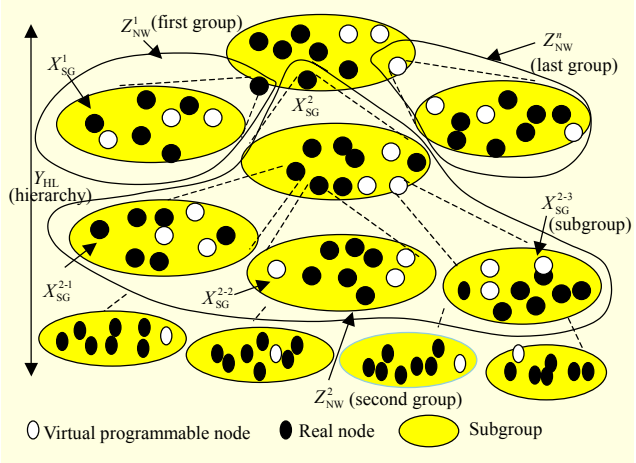

Fig. 1. Hierarchical mobile ad hoc network.

2. Key Generation Using Reduced Key Length with Compact Codes

The McEliece system, based on Reed–Solomon codes, with its compact public key generation and reduced communication is selected for this work [13]. This is an integrated and modified form of the protocol proposed by Berger and others [15]. This simple and efficient code-based process encrypts the data at a much higher rate and generates private and public keys with reduced key length and improved security. Thus, the algorithm is suitable for low-cost devices. According to this algorithm, a public key (PU_i) and private key (PR_i) for the ith mobile node is given by $PU_i = (H, I)$ and $PR_i = (d, G, SP, G_\alpha)$, respectively, where $H = (B_1' \,|\, \ldots \,|\, B_{(q^I-1)}')$. Here, B_i' is the identity code in binary form, $d \in \left(0, 1, 2, \ldots, [(q^I-1)-1]\right)$ is the hamming distance, and $G_\alpha \in (G_\alpha^1, G_\alpha^2, \ldots, G_\alpha^{(q^I-1)})$ is the random primitive element with order smaller than g in the αth iteration of assigning identification marks, discussed in Algorithm 1. According to Definition 1, SP is a random path.

3. Identification Marks to Nodes Using Compact Code Identification Protocols

After generating codes and a key pair, identification marks to nodes are assigned using Algorithm 1. The approach of Aguilar and others of the double-circulant protocol (DCP) for zero-knowledge code-based identification with reduced communication is modified with an encryption/decryption process [43].

Algorithm 1. Identification assignment using DCP with Reed–Solomon codes in quasi-cyclic form.

Goal. Assigning unique identifiers with reduced communication.

Premises. Let R_q and $N_q \in F_q$ be the random numbers, $S^\alpha \in \{1, 2, \ldots, q\}$ is the node's shift operation random number, and $t^\alpha \in \{0,1\}$ is the random number selected by the destination node in αth iteration of assigning the identifier. Suppose J^{G_α} represents the permutation of a randomly selected code from M_{2I} and h is a lightweight hash.

Step 1. Source computes two commitments:

$w_1 = h(J^{G_\alpha^1}(R_\alpha) \ldots h(J^{G_\alpha^{(q^I-1)}}(R_\alpha))$ and $w_2 = h(J^{G_\alpha^1}(R_\alpha \,|\, G) \ldots h(J^{G_\alpha^{(q^m-1)}}(R_\alpha \,|\, G))$ for each node and send it to respective node. Also, $E_{PR_i}(J^{G_\alpha^I}(M_{2I}))$ and $E_{PR_i}(R_\alpha \,|\, J^{G_\alpha^I}(M_{2I}))$ are sent to

destination.

Step 2. Each destination node sends S^{α} to source.

Step 3. ith mobile node source makes S^{α} bits rotation, $e_i = (\text{ROTATION}_{(S)^{\alpha}}(\text{M}_{21}^{\alpha})$ and sends commitment $w_3 = h(J^{G_{\alpha}^{1}}(R_{\alpha} \mid G) + e_i) \ldots h(J^{G_{\alpha}^{(q^{f}-1)}}((R_{\alpha} \mid G) + e_i))$ to destination nodes.

Step 4. Next, each destination node sends t^{α} to source.

Step 5. If $t^{\alpha} = 0$ then source reveals $D_{PR_i}(R_{\alpha} + N_{\alpha})$ and $J^{G_{\alpha}^{1}} \ldots J^{G_{\alpha}^{(q^{f}-1)}}$ else if $t^{\alpha} = 1$ then source reveals $D_{PR_i}(J^{G_{\alpha}^{1}}(R_{\alpha}|G) \ldots J^{G_{\alpha}^{(q^{f}-1)}}(R_{\alpha}|G))$ and $D_{PR_i}(J^{G_{\alpha}^{1}}(e_1) \ldots J^{G_{\alpha}^{(q^{f}-1)}}(e_i))$.

Step 6. Finally, if destination node has sent $t^{\alpha} = 0$ then it will verify w_1 and w_3 else it will verify w_2 and w_3.

4. Collecting Proofs and Locations of Mobile Nodes

The presence of $\text{MN}_{(i,j)}^{(k,n)}$ at some specific location in a random state $s_a^{(x_a,y_a)}$ is identified using Frisbee constructions [44]. The Frisbee process constructs circular overlapping zones to cover a simulation area using an RSSI [45]. The probability of following a particular path through states $s_1^{(x_1,y_1)}$ to $s_p^{(x_p,y_p)}$ is calculated from a Markov chain trajectory. This forms the following state/location matrix:

$$\text{LOC} = \begin{Bmatrix} \left(\left\langle s_1^{(x_1,y_1)} \mid s_2^{(x_2,y_2)} \cdots \mid s_p^{(x_p,y_p)} \right\rangle\right)_1, \\ \left(\left\langle s_1^{(x_1,y_1)} \mid s_2^{(x_2,y_2)} \cdots \mid s_p^{(x_p,y_p)} \right\rangle\right)_2, \\ \vdots \\ \left(\left\langle s_1^{(x_1,y_1)} \mid s_2^{(x_2,y_2)} \cdots \mid s_p^{(x_p,y_p)} \right\rangle\right)_{X_{\text{SG}}} \end{Bmatrix} = \{v^1, v^2, \cdots, v^p\}.$$

Here, v^j is the probability of the ith node following a particular path. Location verification is confirmed through nodes' trust scores. The next step explains the trust management procedure to find the trust scores of mobile nodes so as to verify their locations.

5. Determine Trust Score of Each Mobile Node from its Activities

Trust reflects a node's consistent honesty. Trust management includes trust generation, trust propagation, trust accumulation, and trust application. These elements of trust management are processed through many cycles so as to update trust values at regular intervals. Initially, all nodes are considered to be pure souls; thus, they have complete trust. The trust management steps followed in this work are as follows.

Trust generation. The trust of a mobile node is calculated from a health score. In this work, a health score is calculated from the energy state of a mobile node, the probability of sending positive signals, and the capacity of a node to receive, transmit, or route traffic. Lightweight energy measurement does not require any new hardware and is achieved instantaneously. Now, energy = power × time. An RSSI unit provides the value of "power," and "time" is calculated as the number of bits transmitted divided by the transmission rate. The probability of sending positive signals is measured using connectivity with neighboring nodes, since neighboring nodes know the behavior of the observing node. So, if two nodes were neighbors and had exchanged data, then a positive signal is sent toward the subgroup controller; otherwise, a negative signal is sent. For each node, a subgroup controller calculates the total positive and negative count for every node. The capacity of a node to receive, transmit, or route traffic is measured using QoS parameters, such as goodput, delivery ratio, delay, jitter, coverage, and so on. The three aforementioned factors relating to health score are rated on a scale of 1 (lowest) to 10 (highest) by a subgroup controller. This value of health is directly proportionate to a node's trust level. A subgroup controller then broadcasts this trust score to respective mobile nodes using lightweight encryption.

Trust propagation. During trajectory, each node needs to produce its trust value. A binary trust score for each node will be of the sequence $\{0, 1\}^*$. On coming to a particular state, it will produce a trust value. Thus, the average trust score (A_{TS}) will consist of two factors: the probability that it will come to a specific state (MN_{s_a}) and the length of a sequence of assigned identification binary bits (L_{seq}), as generated by a memory-less sensor source. The average trust score A_{TS} is calculated as $A_{\text{TS}}(\text{MN}_{(i,j)}^{(k,n)}) = \sum_{a=1}^{p} \text{MN}_{s_a} \times L_{\text{seq}}$. To generate the optimized value of A_{TS}, the compact L_{seq} should satisfy the bounds $Q(\text{MN}_{(i,j)}^{(k,n)})/\log c \le L_{\text{seq}} \le Q(\text{MN}_{(i,j)}^{(k,n)})/\log c + 1$, where c is the length of the trust score produced by $\text{MN}_{(i,j)}^{(k,n)}$. Entropy is strongly dependent on the probability of following a particular path. Lesser states indicate less uncertainty. As can be seen from Fig. 2, let us consider the movement of a mobile node from the first subgroup at the kth layer to the jth subgroup at the $(k + 1)$th layer. The states of nodes nearest to the origin will be of a higher probability than those that are away from the

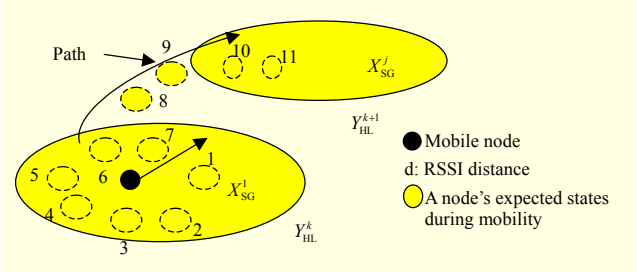

Fig. 2. A_{TS} calculation.

http://dx.doi.org/10.4218/etrij.15.0114.0177

origin. As shown in Fig. 2, the probability of a mobile node moving to either of the positions 1, 2, 3, 4, 5, or 6 is higher than that of it moving to either 7, 8, 9, 10, or 11. In addition, the probability of node 1 will be higher than that of any other neighboring nodes because the direction of the *j*th subgroup at the $(k+1)$th layer can be determined from key generation or the distribution mechanism. Let the probabilities of the paths to be followed (paths 7–11) be 0.9, 0.7, 0.5, 0.25, and 0.1, respectively; and if a source emits a single-bit binary compact code, then A_{TS}= 2.45. Similarly, the A_{TS} scores for 2, 3, 4, 5, 6, and so on can be calculated to be 4.9, 7.35, 9.8, 12.25, 14.7, and so on, respectively.

Trust accumulation. Now, every node is producing its trust score. Hence, the accumulation of a source's trust score is strongly dependent on the reliability of its path. Thus, we can say that $A_{TS}(\mathrm{MN}_{(i,j)}^{(k,n)})$ strongly reflects a path's trust worthiness. Now, a receiver needs to make a decision rule to interpret the original stream with certain probability. Let $B(\mathrm{MN}_{(i,j)}^{(k,n)})$ be the binary code produced and assigned to $\mathrm{MN}_{(i,j)}^{(k,n)}$. The receiver decision rule is a trust function and can be represented as follows:

$$b:\left\{A_{TS}^{s_1^{(x_1,y_1)}}\left(B\left(MN_{(i,j)}^{(k,n)}\right)\right)|\ldots|A_{TS}^{s_p^{(x_p,y_p)}}\left(B\left(MN_{(i,j)}^{(k,n)}\right)\right)\right\}.$$

Using the decision rule function *b*, the receiver will produce an output stream. The selection of code as well as decision by the receiver is based upon the trust score of the path followed. The most trusted path will be preferred. During transmission of trust, strong identification protects the system from various attacks, such as: Sybil, eclipse, collusion, routing, and storage attacks.

Trust application. Many applications are interested in providing services to a network. Each service provided by an application linked to $\mathrm{MN}_{(i,j)}^{(k,n)}$ will have a trust score. This trust score is compared with $A_{TS}(\mathrm{MN}_{(i,j)}^{(k,n)}$. If $A_{TS}(\mathrm{MN}_{(i,j)}^{(k,n)})$ is higher, then access is granted for the provision of service.

V. Attack Analysis

1. Sybil Attack

A probability-based analysis of the proposed system attempts to seek out a mistake that could result in a Sybil attack. A mistake is an alteration of a message or code during transmission. Assume that the probability of a change during transmission is 1–π and that the probability of no change is π. Let, FASA represents the fault acceptance in the case of a Sybil attack. Let FACF describes the fault acceptance when commitments (that is, w_1, w_2, and w_3) are not verified while assigning unique identifications. Let FALV describe the fault acceptance in the case when challenges from/to an outlier ($O_{(h,j)}^{(k,n)}$) are not verified. Let $\mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}$ represent the probability of successful protection from a Sybil attack if the trust score of $\mathrm{MN}_{(i,j)}^{(k,n)}$ is higher than a given threshold. Let NBRC represents that no bit is received during commitment verification in an identification assignment protocol. Let z_{Seq} represent the length of a bit sequence not received during trust propagation, $L_{Seq} \geq z_{Seq}$. Let $\mathrm{NC}_{\mathrm{MN}_{(i,j)}^{(k,n)}}$ represents that no collision will occur for $\mathrm{MN}_{(i,j)}^{(k,n)}$ during its distance verification and mutual authentication.

Now, we have

$$\begin{aligned}P[\mathrm{FASA}] &= P[\mathrm{FASA}\cap\mathrm{FACF}\cap\mathrm{FALV}]\\ &= P[\mathrm{FASA}]\times P[\mathrm{FACF}/\mathrm{FASA}]\\ &\quad\times P[\mathrm{FALV}/\mathrm{FASA}]\\ &\quad\times P[\mathrm{FACF}/\mathrm{FASA}/\mathrm{FALV}].\end{aligned} \tag{1}$$

$$\begin{aligned}P[\mathrm{FASA}] &= [1-(1-\pi)^{L_{Seq}}]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}\\ &\approx\left[1-\left(\frac{1}{2}\right)^{L_{Seq}}\right]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}.\end{aligned} \tag{2}$$

Now, the probability of fault acceptance can be calculated as

$$\begin{aligned}P[\mathrm{FACF}/\mathrm{FASA}] &= P[\mathrm{NBRC}]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}\\ &= \left[1-\left(\frac{1}{2}\right)^{z_{Seq}}\right]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}.\end{aligned} \tag{3}$$

$$P[\mathrm{FALV}/\mathrm{FASA}] = P[\mathrm{NC}_{\mathrm{MN}_{(i,j)}^{(k,n)}}]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}. \tag{4}$$

$$\begin{aligned}&P[\mathrm{FACF}/\mathrm{FASA}/\mathrm{FALV}]\\ &\quad=1-\left(\frac{1}{2}\right)^{L_{Seq}}\times\left[1+\left(\frac{1}{2}\right)^{z_{Seq}}\right]\times \mathrm{PT}_{\mathrm{MN}_{(i,j)}^{(k,n)}}.\end{aligned} \tag{5}$$

By putting (2), (3), (4), and (5) into (1), it can be estimated that *P*[FASA] is very high (≈ 0.9). Thus, the proposed system provides significant protection against Sybil attacks.

2. Eclipse Attack

The proposed system is protected from an eclipse attack because a strong identification mechanism is used to provide unique identifications to nodes. If there is a large delay in the mechanism, then this signals that there are untrusted nodes present. Trust is a fruitful component in this network to maintain relationships among nodes, which in turn reduces the chances of such an attack happening. An eclipse attack is possible if an attacker tries to insert a node to capture a single node (or group of nodes) with an identification mark that has

not yet been assigned to a node. The chances of this type of attack happening can be calculated using the birthday paradox.

If U combinations of L_{Seq} are possible, then $P[\text{FAEA}] = \left(1-\left[\frac{U}{U}\right]\times\left[\frac{U-1}{U}\right]\times\left[\frac{U-2}{U}\right]\times\cdots\times\left[\frac{1}{U}\right]\right)\times \text{PTEA}_{\text{MN}_{(i,j)}^{(k,n)}}$, where FAEA represents the fault acceptance in an eclipse attack and $\text{PTEA}_{\text{MN}_{(i,j)}^{(k,n)}}$ represents the probability of successful protection from an eclipse attack if and only if the trust score of $\text{MN}_{(i,j)}^{(k,n)}$ is higher than a certain threshold. Thus, the number of nodes in each subgroup does not have to exceed ten to make the system stronger against an eclipse attack. Now, if the number of nodes in each subgroup is less than ten, then $P[\text{FAEA}] \geq 0.8$.

3. Desynchronization Attack

The proposed scheme is lightweight because hashing is used at source side; that is, reader side only. Let FADA represent the fault acceptance against a desynchronization attack. Now, we have

$$\begin{aligned} P[\text{FADA}] &= P[\text{FADA}\cap\text{FACF}\cap\text{FALV}] \\ &= P[\text{FADA}]\times P[\text{FACF}/\text{FASA}] \\ &\quad \times P[\text{FALV}/\text{FASA}]\times P[\text{FACF}/\text{FADA}/\text{FALV}]. \end{aligned}$$

However, it has been analyzed in the literature that the blocking verification process increases the chances of a desynchronization attack. Therefore, if w_1 or w_2 is not verified, then $P[\text{FACF}/\text{FASA}] = P[\text{FALV}/\text{FASA}] = P[(\text{FACF}/\text{FADA})/\text{FALV}] \approx 1$. If step 1 and step 2 of Algorithm 1 are verified properly, then the probability of fault acceptance at step 3 is comparatively less, but still acceptable. Here, the number of rotations for the fixed length G is constant. Let ℓ_{DA} represent the number of combinations for L_{DA} bits of G. For each unique value of $2^{L_{\text{DA}}}$, we have

$$P[\text{FADA}] = 1-\left\{\left[\frac{2^{L_{\text{DA}}}}{2^{L_{\text{DA}}}}\right]\times\left[\frac{\left(2^{L_{\text{DA}}}-1\right)}{2^{L_{\text{DA}}}}\right]\times\cdots \times\left[\frac{2^{L_{\text{DA}}}\times\ell_{\text{DA}}+1}{2^{L_{\text{DA}}}}\right]\right\}.$$

The probability of fault acceptance increases with an increase in L_{DA} bits. Even at a lesser bit length (that is, three bits), it provides a higher fault acceptance probability (≈ 0.3). The probability of fault acceptance at step 5 of Algorithm 1 can be calculated as

$$\begin{aligned} &P[\text{FADA}] \\ &= P(\text{Change in } \left(J^{G_\alpha^1}(R_\alpha \mid G)\ldots J^{G_\alpha^{(q^{l-1})}}(R_\alpha|G)\right)\| \\ &\text{Change in } \left(J^{G_\alpha^1}(e_i)\ldots J^{G_\alpha^{(q^{l-1})}}(e_i)\right) \end{aligned}$$

$$\left\{\begin{array}{l} \left\{\left\{1-\left[\frac{2^{J^{G_\alpha^1}(R_\alpha|G)}}{2^{J^{G_\alpha^1}(R_\alpha|G)}}\right]\times\left[\frac{2^{J^{G_\alpha^1}(R_\alpha|G)}-1}{2^{J^{G_\alpha^1}(R_\alpha|G)}}\right]\times\ldots\times\left[\frac{1}{2^{J^{G_\alpha^1}(R_\alpha|G)}}\right]\right\}\times\ldots \\ \times\left\{\left[\frac{2^{J^{G_\alpha^{(q^{l-1})}}(R_\alpha|G)}}{2^{J^{G_\alpha^{(q^{l-1})}}(R_\alpha|G)}}\right]\times\ldots\times\left[1-\left(\frac{1}{2^{J^{G_\alpha^{(q^{l-1})}}(R_\alpha|G)}}\right)\right]\right\} \end{array}\right\}\| \\ \left\{\begin{array}{l} \left\{1-\left[\frac{2^{J^{G_\alpha^1}(e_1)}}{2^{J^{G_\alpha^1}(e_1)}}\right]\times\left[\frac{2^{J^{G_\alpha^1}(e_1)}-1}{2^{J^{G_\alpha^1}(e_1)}}\right]\times\ldots\times\left[\frac{1}{2^{J^{G_\alpha^1}(e_1)}}\right]\right\}\times\ldots \\ \times\left\{\left[\frac{2^{J^{G_\alpha^{(q^{l-1})}}(e_i)}}{2^{J^{G_\alpha^{(q^{l-1})}}(e_i)}}\right]\times\ldots\times\left[1-\left(\frac{1}{2^{J^{G_\alpha^{(q^{l-1})}}(e_i)}}\right)\right]\right\} \end{array}\right\}.$$

The complexity of the system increases with an increase in the number of bits during rotation or permutation combinations. This increase in bits enhances the fault acceptance probability of a de-synchronization attack.

VI. Lightweight Analysis

1. Lightweight Design Analysis

In this section, the proposed protocol is modeled and analyzed using Alloy [10]–[12]. Figure 3 shows an example of an Alloy model used to construct a group in the proposed protocol. In this example model, five nodes form part of a network with a single identity code and a key pair generator. Each node in the network will request the generation of a unique identification. However, identifications shall be assigned to member nodes only. Table 1 shows the delay calculations for such identification generation and key pair generation. Various observations from an analysis of the Alloy model can be seen. These are as follows:

- If each subgroup contains less than 10 member nodes, then it is possible to construct a network with limited delay.
- If the number of member nodes increases from ten to eleven in each subgroup, then a minimum of 170% increase in delay is observed. This brings us to the conclusion that if the

Table 1. Delay calculations in identification and key pair generation.

No. of networks	No. of randomly positioned nodes and members in each network	Time (ms)
1	5/10/11/15/20	62/842/1513/18690/INF
5	5/10/11/15/20	156/1,652/2,823/27,949/INF
10	5/10/11/15/20	141/1,266/2,732/INF/INF
20	5/10/11/15/20	156/843/2,934/INF/INF

ETRI Journal, Volume 37, Number 1, February 2015
http://dx.doi.org/10.4218/etrij.15.0114.0177

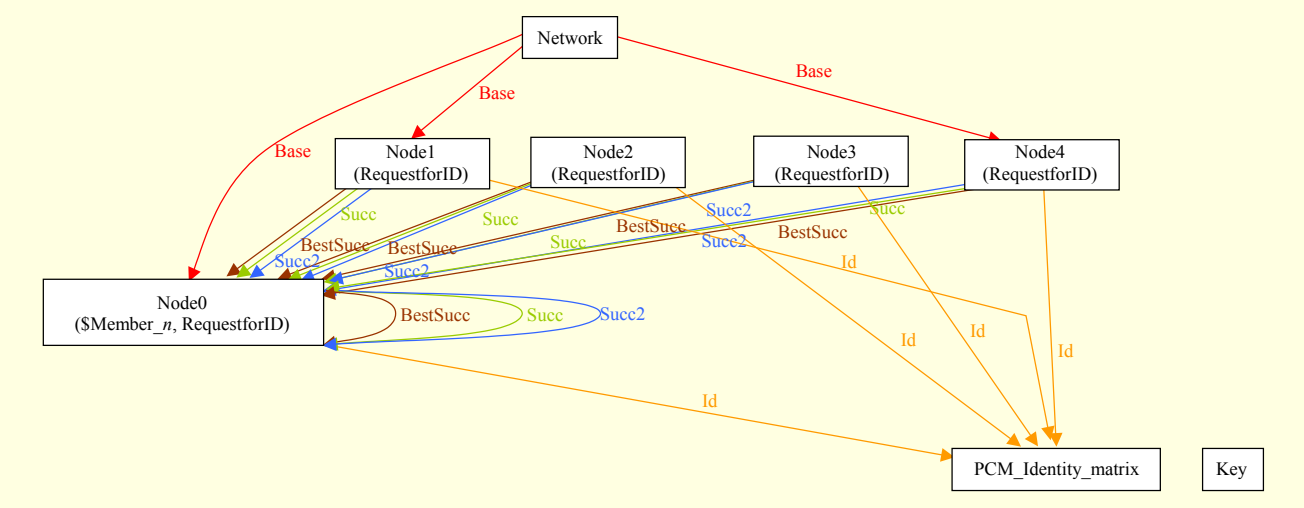

Fig. 3. Counter-example Alloy model of identification generation comprising one network, five randomly positioned nodes, and one key pair generator.

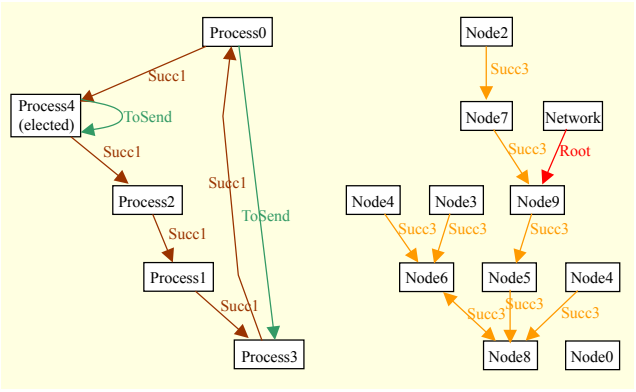

Fig. 4. Counter-example Alloy model to select subgroup controller comprising one network, ten randomly positioned nodes, and five processes.

Table 2. Delay calculations in selecting a subgroup controller.

No. of networks	No. of randomly positioned nodes	No. of processes to select a subgroup controller	Time (ms)
1	10/15/20	5	936/1,950/6,630
1	10/15/20	10	3,456/7,925/INF
5	10/15/20	5	1,232/4,134/6,303
5	10/15/20	10	46,005/50,086/28,139
10	10/15/20	5	1,310/1,794/3,998
10	10/15/20	10	INF/INF/INF
20	10/15/20	5	1,373/2,824/2,402
20	10/15/20	10	INF/INF/INF

number of member nodes in each subgroup is ten, then it is considered to be the best possible solution.

- With an increase in the number of networks, there is an increase in delay, but the delay begins to decrease if the number of nodes in each subgroup is ten.

After subgroup formation, the process of selecting a subgroup controller begins. Figure 4 shows an example of the selecting of a subgroup controller in one network. Five processes are programmed to select a subgroup controller in a network. As shown in Fig. 4, a hierarchical network is constructed. Processes, using both the hierarchical structure and the identification marks of nodes, exchange tokens to select a controller. Every node in this hierarchy is controlled by its parent node. Node 6 and node 8 have a bidirectional link because both are primary subgroups in the upper-most layer. Table 2 shows the delay calculations when five or ten processes are run to select a subgroup controller. The results show that if five processes are selected to elect a subgroup controller, then there is less delay compared to when ten processes are selected. The delay increases with an increase in the number of nodes, networks, or processes. However, if there are more than ten networks, ten nodes, and ten processes, then it is not feasible to select a subgroup controller within a limited time period. Thus, if the number of networks or nodes increases, then a five processes–based subgroup controller election is preferred.

Figure 5 shows a prototype model of the remaining components in the proposed methodology. In this example, five nodes have location coordinates $(x_a, y_a), a \in \{0,1,...,4\}$. The locations of the five nodes are observed and verified through these coordinates. As discussed in section IV, a trust management procedure is used to find the trust scores of mobile nodes so as to verify these locations. A trust score is generated at a single point, which may be at a subgroup

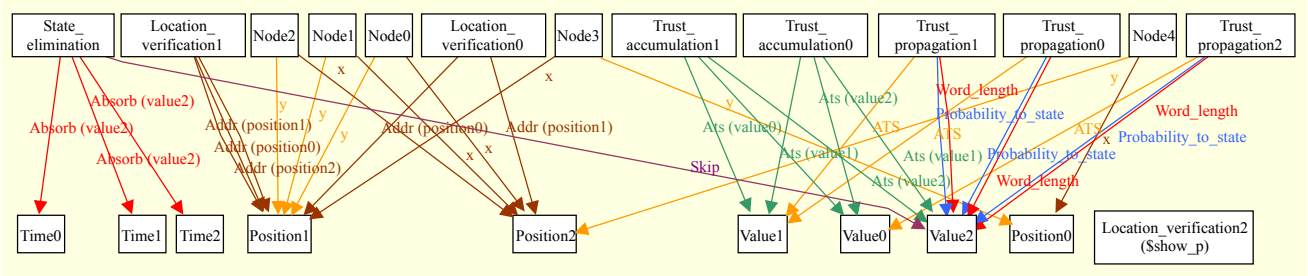

Fig. 5. Counter-model example of proposed system comprising one network and five randomly positioned nodes.

Table 3. Attack detection delay comparisons.

Attack	Detection time in proposed system	Average detection time in other systems
Sybil	≤ 310 msec	≤ few seconds [46]–[48]
Eclipse	≤ 500 msec	Few msec to seconds [38]–[39]
Desynchronization	≤ 190 msec	≤ few seconds [49]

```
( … (MemberNoConflict = (Interested s a r) :- (!Conflicted s r) (ActController
a) (LocalGroup r))
(ControllerNoConflict = (Interested s a r) :- (!Conflicted s r) (ActController a)
(LocalGroup r))
(ControllerAssigned = (NotInterested s a r) :- (Permitted s r) (ActController a)
(GlobalGroup r))
(ControllerConflict = (DenyAccess s a r) :- (Conflicted s r) (ActController a)
(Hierarchy r))
(ControllerNoConflict = (Interested s a r) :- (Permitted s r) (AssignID a)
(LocalGroup r))
(ControllerAssigned = (Interested s a r) :- (Permitted s r) (AssignID a)
(LocalGroup r)) … )
```

Fig. 6. Counter example of controller policy analysis in proposed scheme using Margrave.

controller or central point. This trust score can be propagated or accumulated through multiple paths. State elimination is a process designed to shorten the path of a trust propagation without affecting the trust score value. This process absorbs or skips the states that do not affect the $A_{\mathrm{TS}}(\mathrm{MN}_{(i,j)}^{(k,n)})$ calculation. Table 3 shows the analysis of delays in detecting Sybil, eclipse, and desynchronization attacks. The results show that the proposed system detects the attacks in a very short time as compared to other systems.

2. Lightweight-Policy Analysis

In the proposed scheme, the source node, destination node, subgroup member, subgroup controller, intermediate node, primary subgroup, next layer subgroup, and so on are the actors in the scheme, and all actors are playing different roles within the scheme. For example, member nodes are a part of groups or subgroups. These groups or subgroups are controlled by a subgroup controller. Every subgroup controller can generate a trust score, pass a trust score to a neighboring subgroup controller, collect and verify the locations of nodes, and so on. In contrast, each intermediate node does not play the role of a controller. However, an intermediate node with a high health score can act as a next-layer subgroup controller. In addition, an intermediate node can collect or pass the $A_{\mathrm{TS}}(\mathrm{MN}_{(i,j)}^{(k,n)})$ to neighboring nodes. Passing states are absorbed when they are not affecting the results of trust propagations. Hence, one node can play multiple roles. There should not be any conflict of interest among any member in playing its role. A well-defined policy can avoid such conflicts. For example, Fig. 6 shows an example of subgroup controller policies. In this example, six policies are designed and implemented with the help of a subject (s), an action (a), and a resource (r). The subject (s) is an actor in a network and can perform various actions (FormSubgroup, AssignID, RetrieveID, ActController, and so on) on resources (LocalGroup, GlobalGroup, Hierarchy, Network, and so on). In the first policy, all interested and permitted members are allowed to act as controller. In the next three policies, interested and permitted controllers can control a local subgroup, a global subgroup (a subgroup from another network), or a hierarchy (if the controller is a member of a root subgroup). A permitted and non-conflicting controller can assign unique identification marks to LocalGroup members. Similarly, policies for other actors are designed and analyzed in Margrave [8]–[9]. The results show that there is no conflict in the role of any actor.

VII. Conclusion

This paper examined a sequence of steps in RFID sensor–based MANETs to construct a lightweight secure environment. An ECC in a code-based cryptosystem is used to analyze identification, key generation, authentication, and location verification protocols in lightweight cryptography. An Alloy analyzer is used to construct a lightweight model for the proposed sequence of steps to integrate the protocols. This analysis shows that ten nodes and five processes in a subgroup are used to construct a network that detects Sybil, eclipse and

desynchronization attacks in only a few milliseconds. Network construction is administered through actors. Subgroup members, subgroup controllers, source nodes, destination nodes, and intermediate nodes are the actors that control the network's activities. A Margrave policy analysis shows that there is no conflict of interest among the different roles of the actors.

References

[1] S.A. Anson and M. Ilyas, "*RFID Handbook: Application Technology, Security and Privacy*," Boca Raton, USA: CRC, 2008, pp. 35–64.

[2] L. Zhang and Z. Wang, "Integration of RFID into Wireless Sensor Networks: Architectures, Opportunities and Challenging Problems," *Proc. Int. Conf. Grid Cooperative Comput. Workshops*, Hunan, China, Oct. 21–23, 2006, pp. 463–469.

[3] C. Englund and H. Wallin, "*RFID in Wireless Sensor Network*," M.S. thesis, Commun. Syst. Group, Dept. Signals Syst., Chalmers University of Technology, Goteborg, Sweden, Apr. 2004.

[4] R.B. Ferguson, *Gentag Patent Adds RFID Sensor Network Feature to Mobile Devices*, Mobile News and Reviews, eWeek, Dec. 2006. Accessed Nov. 2012. http://www.eweek.com/c/a/Mobile-and-Wireless/Gentag-Patent-Adds-RFID-Sensor-Network-Feature-to-Mobile-Devices

[5] A. Kumar, A. Aggarwal, and T. Charu, "Efficient Hierarchical Threshold Symmetric Group Key Management Protocol for Mobile Ad Hoc Networks," *Int. Conf. Contemporary Comput.*, Noida, India, Aug. 6–8, 2012, pp. 335–346.

[6] A. Kumar, K. Gopal, and A. Aggarwal, "Outlier Detection and Treatment for Lightweight Mobile Ad Hoc Networks," *Int. Conf. Heterogeneous Netw. Quality, Rel., Security Robustness*, Greader Noida, India, Jan. 11–12, 2013, pp. 750–763.

[7] G-S. Ahn et al., "Funneling-MAC: A Localized, Sink-Oriented MAC for Boosting Fidelity in Sensor Networks," *ACM Int. Conf. Embedded Netw. Sensor Syst.*, Boulder, CO, USA, Nov. 1–3, 2006, pp. 293–306.

[8] T. Nelson et al., "The Margrave Tool for Firewall Analysis," *USENIX Large Installation Syst. Admin. Conf.*, San Jose, CA, USA, Nov. 7–12, 2010, pp. 1–18.

[9] S. Saghafi, T. Nelson, and D.J. Dougherty, "Geometric Logic for Policy Analysis," *Int. Workshop Autom. Reasoning Security Softw. Verification*, Lake Placid, NY, USA, June 9, 2013, pp. 12–20.

[10] D. Jackson, "*Software Abstractions: Logic, Languages, and Analysis*," Cambridge, MA: MIT Press, 2006.

[11] D. Jackson, "*A Micromodels of Software: Lightweight Modelling and Analysis with Alloy*," Software Design Group, MIT Lab Manual, Cambridge, USA: MIT, Feb. 2002, pp. 1–58.

[12] D. Jackson, "Alloy: A Lightweight Object Modelling Notation," *ACM Trans. Softw. Eng. Methodology*, vol. 11, no. 2, pp. 256–290.

[13] R.J. McEliece, "A Public Key Cryptosystem Based on Algebraic Coding Theory," Jet Propulsion Lab., CA, USA, Deep Space Network Progress, Report 42–44, Feb. 1978, pp. 114–116.

[14] T. Eisenbarth et al., "MicroEliece: McEliece for Embedded Devices," *Proc. Cryptographic Hardware Embedded Syst.*, Lausanne, Switzerland, Sept. 6–9, 2009, pp. 49–64.

[15] T.P. Berger et al., "Reducing Key Length of the McEliece Cryptosystem," *Progress Cryptology-AFRICACRYPT*, Gammarth, Tunisia, June 21–25, 2009, pp. 77–97.

[16] L. Minder and A. Shokrollahi, "Cryptanaysis of the Sidelnikov Cryptosystem," *EUROCRYPT*, Barcelona, Spain, May 20–24, 2007, pp. 347–360.

[17] C. Faure and L. Minder, "Cryptanalysis of the McElice Cryptosystem over Hyper Elliptic Codes," *Int. Workshop Algebraic Combinatorial Coding Theory*, Pamporovo, Bulgaria, June 16–22, 2008, pp. 99–107.

[18] J.K. Gibson, "The Security of the Gabidulin Public Key Cryptosystem," *Adv. Cryptology, EUROCRYPT*, Saragossa, Spain, May 12–16, 1996, pp. 212–223.

[19] R. Overbeck, "Structural Attacks for Public Key Cryptosystems Based on Gabidulin Codes," *J. Cryptology*, vol. 21, no. 2, Apr. 2008, pp. 280–301.

[20] V.M. Sidelnikov and S.O. Shestakov, "On Insecurity of Cryptosystems Based on Generalized Reed-Solomon Codes," *Discrete Math. Appl.*, vol. 2, no. 4, Jan. 1992, pp. 439–444.

[21] N. Sastry, U. Shankar, and D. Wagner, "Secure Verification of Location Claims," *ACM Workshop Wireless Security*, San Diego, CA, USA, Sept. 28, 2003, pp. 1–10.

[22] S. Capkun, M. Cagalj, and M. Srivastava, "Secure Localization with Hidden and Mobile Base Stations," *IEEE INFOCOM*, Barcelona, Spain, Apr. 23–29, 2006, pp. 1–10.

[23] M. Talasila, R. Curtmola, and C. Borcea, "LINK: Location Verification through Immediate Neighbors Knowledge," *Int. Conf. Mobile Ubiquitous Syst.: Comput. Netw. Serv.*, Sydney, Australia, Dec. 6–9, 2010, pp. 210–223.

[24] Y. Wei, Z. Yu, and Y. Guan, "Location Verification Algorithms for Wireless Sensor Networks," *IEEE Int. Conf. Distrib. Comput. Syst.*, Toronto, Canada, June 25–27, 2007, pp. 938–950.

[25] D. Molnar, A. Soppera, and D. Wagner, "A Scalable Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags," *Sel. Areas Cryptography*, Kingston, Canada, Aug. 11–12, 2005, pp. 276–290.

[26] Y. Tian, G. Chen, and J. Li, "A New Ultra lightweight RFID Authentication Protocol with Permutation," *IEEE Commun. Lett.*, vol. 16, no. 5, May 2012, pp. 702–705.

[27] U. Mujahid et al., "Cryptanalysis of Ultra lightweight RFID Authentication Protocol," IACR Cryptology ePrint Archive, Report 2013/385, 2013.

[28] A. Juels, "Yoking-Proofs for RFID Tags," *IEEE Conf. Pervasive Comput. Commun.*, Orlando, FL, USA, 2004, pp. 138–143.

[29] L. Lamport, "Constructing Digital Signatures from a One Way

Function," SRI Int., CA, USA, Technical Report CSL–98, 1979.
[30] J. Saito and K. Sakurai, "Grouping Proof for RFID Tags," *Int. Conf. Adv. Inf. Netw. Appl.*, Taipei, Taiwan, vol. 2, Mar. 28–30, 2005, pp. 621–624.
[31] S. Piramuthu, "On Existence Proofs for Multiple RFID Tags," *ACS/IEEE Int. Conf. Pervasive Services*, Lyon, France, June 26–29, 2006, pp. 317–320.
[32] J.-S. Cho et al., "Enhanced Yoking Proof Protocols for RFID Tags and Tag Groups," *Int. Conf. Adv. Inf. Netw. Appl.*, Okinawa, Japan, Mar. 25–28, 2008, pp. 1591–1596.
[33] C.-F. Lee et al., "Anonymous RFID Yoking Protocol Using Error Correction Codes," *Int. Conf. Radio Freq. Identification Syst. Security*, Singapore City, Singapore, Feb. 2010, pp. 147–157.
[34] H.-Y. Chien, "Combining Rabin Cryptosystem and Error Correction Codes to Facilitate Anonymous Authentication with Un-traceability for Low-End Devices," *Comput. Netw.*, vol. 57, no. 14, Oct. 2013, pp. 2705–2717.
[35] T. Cholez et al., "Detection and Mitigation of Localized Attacks in a Widely Deployed P2P Network," *Peer-to-Peer Netw. Appl.*, vol. 6, no. 2, June 2013, pp. 155–174.
[36] P.W.L. Fong, "Preventing Sybil Attacks by Privilege Attenuation: A Design Principle for Social Network Systems," *IEEE Symp. Security Privacy*, Berkely, CA, USA, May 2011, pp. 263–278.
[37] G. Danezis et al., "Sybil-Resistant DHT Routing," *European Symp. Res. Comput. Security*, Milan, Italy, 2005, pp. 305–318.
[38] A. Singh et al., "Eclipse Attacks on Overlay Networks: Threats and Defenses," *INFOCOM*, Barcelona, Spain, Apr. 2006, pp. 1–12.
[39] A. Singh et al., "Defending against Eclipse Attacks on Overlay Networks," *ACM Special Interest Group Operating Syst. European Workshops*, Leuven, Belgium, Sept. 19–22, 2004, Article 21, pp. 1–6.
[40] U. Mujahid, M. Najam-ul-islam, and J. Ahmed, "Ultra lightweight Cryptography for Passive RFID System," IACR Cryptology ePrint Archive, Report 2013/847, 2013.
[41] H. Kim, "Desynchronization Attack on Hash-Based RFID Mutual Authentication Protocol," *J. Security Eng.*, vol. 9, no. 4, Aug. 2012, pp. 357–365.
[42] T.-V. Deursen and S. Radomirovic, "Security of RFID Protocols – A Case Study," *Electron. Notes Theoretical Comput. Sci.*, vol. 244, Aug. 2009, pp. 41–52.
[43] C. Aguilar, P. Gaborit, and J. Schrek, "A New Zero-Knowledge Code Based Identification Scheme with Reduced Communication," *Inf. Theory Workshop*, Paraty, Brazil, Oct. 16–20, 2011, pp. 648–652.
[44] A. Cerpa et al., "Habitat Monitoring Application Driver for Wireless Communication Technology," *ACM SIGCOMM Workshop Data Commun.*, San Jose, Costa Rica, vol. 31, no. 2, Aug. 27–31, 2001, pp. 20–41.
[45] J. Zhan, L.-X. Wu, and Z.-J. Tang, "Research on Ranging Accuracy Based on RSSI of Wireless Sensor Network," *Int. Conf. Inf. Sci. Eng.*, Hangzhou, China, Dec. 4–6, 2010, pp. 2338–2341.
[46] N. Dutta and S. Chellappan, "A Time-Series Clustering Approach for Sybil Attack Detection in Vehicular Ad Hoc Networks," *Int. Conf. Adv. Veh. Syst., Technol. Appl.*, Nice, France, July 21–26, 2013, pp. 35–40.
[47] B. Lee, E. Jeong, and I. Jung, "A DTSA (Detection Technique against a Sybil Attack) Protocol Using SKC (Session Key Based Certificate) on VANET," *Int. J. Security its Appl.*, vol. 7, no. 3, May 2013, pp. 1–10.
[48] T. Zhou et al., "P^2DAP – Sybil Attacks Detection in Vehicular Ad Hoc Networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 3, Mar. 2011, pp. 582–594.
[49] B. Niu, X. Zhu, and H. Li, "An Ultra lightweight and Privacy Preserving Authentication Protocol for Mobile RFID Systems," *IEEE Wireless Commun. Netw. Conf.*, Shanghai, China, Apr. 7–10, 2013, pp. 1864–1869.

Adarsh Kumar received his ME degree in software engineering from Thapar University, Patiala, Punjab, India, in 2003. Since 2003, he has been with the Department of Computer Science Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, India, where he is now an assistant professor. His main research interests are cryptography, network security, and ad hoc networks.

Krishna Gopal received his BTECH degree in electrical engineering from the Department of Electrical Engineering, IIT, Madras, India, in 1966 and his MS and PhD degrees in engineering from the REC Kurukshetra, India, in 1972 and 1979, respectively. Since 2011, he has been working as a dean (Academic & Research) with JIIT, Noida, India. He has forty-five years of teaching and research experience. He is a member of various professional bodies, such as the Life Member System Society of India, the Indian Society for Technical Education, and the IEEE.

Alok Aggarwal received his BTECH and MTECH degrees in computer science engineering from the Department of Computer Science, Kurukshetra University, India, in 1995 and 2001, respectively and his PhD degree in engineering from IIT, Roorkee, India, in 2010. From 2009 to 2012, he worked for the Jaypee Institute of Information Technology, Noida, India. Since 2012, he has been with the JP Institute of Engineering and Technology, Meerut, India, where he is now both a professor and a director. His main research interests are wired/wireless networks, security, and coding theory.