Received June 24, 2020, accepted June 29, 2020, date of publication July 9, 2020, date of current version July 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3008230

An Enhanced Authentication Protocol for RFID Systems

MEHDI HOSSEINZADEH[1,2], OMED HASSAN AHMED[3], (Member, IEEE), SARKAR HASAN AHMED[4], CUONG TRINH[5], NASOUR BAGHERI[6,7], SARU KUMARI[8], JAN LANSKY[9], AND BAO HUYNH[10]

[1]Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam
[2]Health Management and Economics Research Center, Iran University of Medical Sciences, Tehran 16668-87635, Iran
[3]Department of Information Technology, University of Human Development, Sulaymaniyah 00964, Iraq
[4]Network Department, Sulaimani Polytechnic University, Sulaymaniyah 46001, Iraq
[5]Artificial Intelligence Laboratory, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam
[6]Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran 16788-15811, Iran
[7]School of Computer Science (SCS), Institute for Research in Fundamental Sciences (IPM), Farmanieh Campus, Tehran 19538-33511, Iran
[8]Department of Mathematics, Chaudhary Charan Singh University, Meerut 250004, India
[9]Department of Computer Science and Mathematics, Faculty of Economic Studies, University of Finance and Administration, 101 00 Prague, Czech Republic
[10]Faculty of Information Technology, Ho Chi Minh City University of Technology (HUTECH), Ho Chi Minh City 700000, Vietnam

Corresponding authors: Bao Huynh (hq.bao@hutech.edu.vn)

This work was supported by using institutional support for long-term conceptual development of research organization University of Finance and Administration.

ABSTRACT In this paper, we analyse the security of two mutual authentication protocols that have been recently proposed by Gao *et al.* (IEEE Access, 7:8376–8384, 2019), a hash-based protocol and a Rabin public key based protocol. Our security analysis clearly shows important security pitfalls in these schemes. More precisely, in each protocol, we introduce efficient approaches to desynchronize the tag and the reader/server. The proposed attacks are almost deterministic and the complexity of each attack is a session for the hash-based and three sessions for Rabin public key based protocol. In addition, in the case of the hash-based protocol, we extend the proposed desynchronization attack to a traceability attack in which the adversary can trace any given tag based on the proposed attack with probability of almost one. In the case of Rabin public key based protocol, we extend the proposed desynchronization attack to a tag impersonation attack with the success probability of one. Besides, we propose an enhanced version of the Rabin public key based protocol to provide a secure authentication between the tag and the reader. We evaluate the security of the proposed protocol formally using the Scyther tool and also in Real-or-Random model.

INDEX TERMS IoT, RFID, mutual authentication, security analysis, desynchronization, traceability, impersonation, real-or-random model.

I. INTRODUCTION

These days many objects have the capability of communicating with other objects or jointing a communication-network, e.g. internet, to transfer or receive data. The Internet of Things (IoT) is a proper infrastructure to employ this capability to enhance the quality of our daily life. IoT is a novel paradigm that has gained popularity in the last decade. The terms started to be used by Auto-ID Labs, which are the leading global network of academic research laboratories in the field of networked Radio-Frequency Identification (RFID). At the beginning, the term ''things'' made only reference to simple items, e.g. RFID tags. Nevertheless, the term is much wider, including sensors, actuators, mobile phones and everyday things like home appliances, food packages, cloths, paper documents and so on. Nowadays, IoT has many subfields such as internet of vehicles, internet of sensors, internet of energy, Machine to Machine (M2M) communications which combined with new advances in artificial intelligence and machine learning, e.g. deep learning, and big data analytic expand the IoT vision [1]. For example, the number of connected devices around the world in IoT based services is expected to be 125 billion connected devices by 2030 [2], [3]. Those huge population of devices can be categorized in different IoT based applications and services. Some samples are the IoT-based services that are used in Smart-City domain, in the industry domain for example in factories and issues of logistics of resources and products, in the healthcare

The associate editor coordinating the review of this manuscript and approving it for publication was Noor Zaman.

domain, and many others [4], [5]. However, the exponential growth in the number of smart devices connected to IoT, associated with various IoT-based smart applications and services, raises other challenges such as compatibility and interoperability which could affect the sustainability, stability and resiliency of IoT services [5]. On the other hand, this extremely huge network of things, combined with big data analytics, also provides a unique opportunity for both disaster management systems and disaster-related authorities (emergency responders, police, public health, fire departments and many other critical services providers) to acquire state-of-the-art assistance and improved insights for accurate and timely decision-making [6].

Although the number of devices connected to the Internet is huge (already greater than the worldwide population [3]), advances on IoT architecture, protocols and adversary models to cater IoT devices are still needed.

From all the technologies immersed in IoT, Radio-frequency identification (RFID) is one of the leading ones due to its maturity, low cost and strong support from the industry [7]. RFID is a prominent automated identification technology based on radio frequencies with a wide range of advantages and applications, including read and write the data of an item (i.e., person, animal or product). Although important advances have occurred in the last years, the design of secure protocols for a constrained environment, e.g. RFID and IoT, is still a challenge [8], because most of those devices that are connected by RFID and IoT are very constrained and we may not be able to use common security solutions for their communications. Hence, one should design a lightweight protocol that meets such environment constraints. Many proposed, broken and enhanced protocols for these environments are evidences of such attempts. To highlight the importance of the security concerns in this area, it worth to mention the current NIST (its Computer Security Resource Center) competition for low-cost cryptography (LWC) [9].

In RFID systems, to achieve intended security objectives, readers and tags may employ authentication protocols in order to achieve one-end or mutual authentication, in a mutual authentication protocol readers and tags are authenticating each other while in one-end authentication only one party authenticates the other party. Authentication protocols commonly exchange a number of messages between the involved entities. We can also categorize authentication protocols according to their connection capabilities with the back-end server. Those with a permanent connection to the back-end server, and those without – or with only an off-line connection – are the two main categories. To provide a secure authentication between the reader and the tag, Tan *et al.* proposed several protocols. However, later analyses [10] pointed out their security pitfalls. Besides that, Wang and Ma [11] showed other security weaknesses of Tan *et al.* protocol [12] and also proposed a server-less protocol. However, recently Gao *et al.* also analysed this server-less protocol and have shown that it suffers from traceability attack [13]. Besides, they proposed two authentication protocols respectively based on hash function and Rabin public key cryptography. However, any new security protocol should be carefully evaluated by independent researchers before employment in any real-life application. Hence, in this paper, we consider the security of the proposed protocols by Gao *et al.* in depth, as the first third party analysis of these protocols to the best of our knowledge.

A. OUR CONTRIBUTION

This article's main contributions are as follows:

1) We analyse the security of the hash-based mutual authentication protocol proposed by Gao *et al.* and show that, given any tag, it is possible to desynchronize it from the reader with the complexity of just a session of the protocol. We also show that any desynchronized tag will be always traceable by the adversary.
2) We also analyse the security of the Rabin public key cryptography based mutual authentication protocol proposed by Gao *et al.* and show that, given any tag, it is possible to desynchronize it from the reader with the complexity of just three sessions of the protocol. We also show that the adversary can impersonate any desynchronized tag with the probability of one at any desired time.
3) We propose an enhanced version of the Rabin public key based protocol to provide a secure authentication between the tag and the reader.
4) We evaluate the security of the proposed protocol formally using the Scyther tool and also in Real-or-Random model.

B. PAPER ORGANIZATION

The paper is organized as follows. Section II is dedicated to the description of the Gao *et al.* mutual authentication protocols [11]. In Section III, the security analysis of these protocols is provided. Section IV is dedicated to the proposed protocol. The security analysis of the proposed protocol and its comparison is performed in Section V. Finally, our closing remarks and recommendations to design a secure protocol are presented in Section VI.

II. Gao *et al.* MUTUAL AUTHENTICATION PROTOCOLS DESCRIPTION

Gao *et al.* [13] have analysed the security of the server-less mutual authentication protocol proposed by Wang and Ma [11] and have shown that this protocol is vulnerable to tracing attacks. In Wang and Ma [11] protocol, to reduce the search time in the reader/server, the tags transmit $h(f(r_i, t_j))_m$ which is used as a mechanism to improve the search time for the reader. However, Gao *et al.* have shown that this data can be used as a measure to trace the tag. To overcome these attacks, Gao *et al.* proposed two new protocols, a hash-based dynamic grouping indexing protocol and Rabin public key cryptography based protocols. In both protocols, to avoid traceability, the tags parameters are updated after

TABLE 1. Notations used in this paper.

Notation	Description
R	the RFID reader
R/S	the RFID reader+server
S	the server
T_i	i-th RFID tag
K_m, K_h, K_l	the secret keys of the tag
IDS	the tag's identifier
g_i	the i-th group index
K_P^S	the public key of the server, in Rabin public key crypto system, i.e. $(p \times q)$ where p and q are large prime numbers
K_S^S	the private key of the server, in Rabin public key crypto system, i.e. p and q
K_P^T	the public key of the tag, in Rabin public key crypto system
K_S^T	the private key of the tag, in Rabin public key crypto system
n_i	i-th random number
$\mathcal{L}_i$	the access list for $\mathcal{R}_i$
n	number of entries in the access list $\mathcal{L}_i$ or group g_i
m	number of entries list $\mathcal{L}_i$ or group indexes
$h(\cdot)$	one-way hash function
$\mathcal{X}^{old}$	the old record of variable $\mathcal{X}$
$\mathcal{X}^{new}$	the new record of variable $\mathcal{X}$
$\mathcal{X}^i$	the value of parameter $\mathcal{X}$ on session i
ℓ	the output length of $h(.)$
$A \rightarrow B$	sending a message from A to B

each successful run of protocols. In addition, in both protocols, to avoid desynchronization attacks, the reader/server keeps a record of the old tag's data also. We now give a brief description of these schemes, but we urge the reader to consult the original paper for further details [13]. Throughout this paper, we use the notations indicated in Table 1.

A. THE HASH-BASED DYNAMIC GROUP-INDEXING AUTHENTICATION PROTOCOL OF Gao et al.

Considering the weaknesses and advantages of Wang and Ma protocol [11], Gao *et al.* proposed a hash-based authentication protocol with dynamic grouping index. In this protocol, tags are indexed in m groups and each group includes n tags. The group index is denoted by g_i. Each tag has three secret keys that are denoted by K_h, K_m and K_l respectively. To avoid traceability attack, the tags parameters are renewal at the end of each successful session of the protocol. In addition, to avoid desynchronization attacks, the reader keeps a history of the old parameters of the tag. As it is depicted in Figure 1, the protocol process runs as follows between reader + server R/S and the tag T_i:

1) R/S generates a random number n_1 and sends it to T_i.
2) T_i replies with $A = h(g_i)$ and $B = h(K_h \oplus g_i \oplus n_1)$.
3) Given $A = h(g_i)$, R/S finds the group index of the tag and tries to find a match for T_i among the tags in this group, based on the received $B = h(K_h \oplus g_i \oplus n_1)$. If R/S could find a match then T_i is authenticated successfully otherwise the authentication fails. Assuming that T_i has been authenticated by R/S, it generates a random number n_2, calculates $C = h(K_m \oplus n_2)$ and sends them to T_i.
4) T_i evaluates the received message C to authenticate R/S. Assuming that T_i has authenticated R/S, it generates a random number n_3, calculates $D = h(K_l \oplus n_2) \oplus n_3$ and sends them to R/S. T_i also updates its group index and secrets.
5) R/S evaluates the received message D to authenticate T_i and updates the group index and the tag's secret keys.
6) In the updating step for T_i, it updates its secret key K_x, for $x \in \{h, m, l\}$, as $K_x = h(K_x \oplus n_2 \oplus n_3)$ and updates its grouping index to g_j, where $j = n_3 \ mod \ m$, where m is the number of groups in the database of the reader.
7) In the updating step for R/S, it first stores the current set of the keys and g_i which has been used to authenticate the tag as *old* set of the information for T_i, i.e. $(K_h^{old}, K_m^{old}, K_l^{old}, g_i^{old})$ then it calculates secret key K_x^{new}, for $x \in \{h, m, l\}$, as $K_x^{new} = h(K_x^{old} \oplus n_2 \oplus n_3)$ and new grouping index g_j^{new}, where $j = n_3 \ mod \ m$, where m is the number of groups in the database of the reader.

B. Gao et al. RABIN PUBLIC KEY BASED AUTHENTICATION PROTOCOL

Gao *et al.* also proposed a public key cryptography based protocol and for this purpose they used the Rabin crypto system. Rabin is a quadratic residue based asymmetric encryption system based on the difficulty of factoring large numbers. In this crypto system, the private key is a pair of large prime numbers (p, q) and the public key is $n = p \times q$. To encrypt a message m, it is enough to compute $m^2 \ mod \ n$ and decryption is done using extended Euclidean algorithm and Chinese Remainder Theorem (CRT). However, decrytpted message is not unique and it is one-to-four map (in general) so it is necessary to have a rule to choose the correct solution in decryption [14]. In the Gao *et al.*'s protocol, the public key and the secret key of T_i are denoted by K_P^T and K_S^T respectively and the public key and the secret key of the server S are denoted by K_P^S and K_S^S respectively. As it is depicted in Figure 2, the protocol process runs as follows between the reader R, the server S and the tag T_i:

1) R sends a *Hello* request to T_i.
2) T_i generates a random number n_1, computes $m_1 = n_1 \oplus K_h$, encrypts n_1 and $(IDS \oplus n_1)$ with K_P^S as $A = n_1^2 \ mod \ K_P^S$ and $B = (IDS \oplus n_1)^2 \ mod \ K_P^S$ respectively, encrypts m_1 with K_P^T as $C = m_1^2 \ mod \ K_P^T$ and sends (A, B, C) to R.
3) R forwards the received (A, B, C) to the server S.

Reader+Server R/S $(K_h^{old}, K_m^{old}, K_l^{old}, g_i^{old}, K_h^{new}, K_m^{new}, K_l^{new}, g_i^{new})$		**Tag** T (K_h, K_m, K_l, g_i)
Generates n_1	$\xrightarrow{n_1}$	
	$\xleftarrow{A,B}$	Computes $A = h(g_i)$ and $B = h(K_h \oplus g_i \oplus n_1)$
Verifies the received B based on the received g_i to authenticate T Generates n_2 and calculates $C = h(K_m \oplus n_2)$		
	$\xrightarrow{C,n_2}$	Extracts n_2 and verifies C to authenticate R/S
Extracts n_3	$\xleftarrow{D}$	Generates n_3 and computes $D = h(K_l \oplus n_2) \oplus n_3$
R/S is updated		T is updated
Updating: $K_x^{old} = K_x^{new}$, for $x \in \{m, h, l\}$ $K_x^{new} = h(K_x^{new} \oplus n_2 \oplus n_3)$, for $x \in \{m, h, l\}$ $g_i^{old} = g_i^{new}$ $g_i^{new} = g_j (t_i \in \forall g_j, j = n_3 \bmod m)$		Updating: $K_x = h(K_x \oplus n_2 \oplus n_3)$, for $x \in \{m, h, l\}$ $g_i = g_j$ $t_i \in \forall g_j, j = n_3 \bmod m$

FIGURE 1. The hash based dynamic grouping authentication protocol of Gao *et al.*

Reader+Server R/S $(K_h^{old}, K_m^{old}, K_l^{old}, IDS^{old}, K_h^{new}, K_m^{new}, K_l^{new}, IDS^{new}, K_S^T, K_P^T, K_S^S, K_P^S)$		**Tag** T $(K_h, K_m, K_l, IDS, K_S^T, K_P^T, K_P^S)$
	$\xrightarrow{Hello}$	
	$\xleftarrow{A,B,C}$	Generates n_1, computes $m_1 = n_1 \oplus K_h$, $A = n_1^2 \bmod K_P^S$, $B = (IDS \oplus n_1)^2 \bmod K_P^S$ and $C = m_1^2 \bmod K_P^T$
Decrypts A and B using K_S^S Extracts IDS as $IDS = n_1 \oplus IDS \oplus n_1$ Finds the tags private key K_S^T and its secret keys K_h, K_m and K_l Decrypts C using K_S^T and verifies it to authenticate the tag Computes $D = \{(n_1 \oplus K_m)^2 \bmod K_P^T\} \oplus K_l$ R/S is updated	$\xrightarrow{D}$	Verifies D T is updated
Updating: $K_x^{old} = K_x^{new}$, for $x \in \{m, h, l\}$ $K_x^{new} = K_x^{new} \oplus n_1$, for $x \in \{m, h, l\}$ $IDS^{old} = IDS^{new}$ $IDS^{new} = IDS^{new} \oplus n_1$		Updating: $K_x = K_x \oplus n_1$, for $x \in \{m, h, l\}$ $IDS = IDS \oplus n_1$

FIGURE 2. The Rabin public key based authentication protocol of Gao *et al.*

4) S decrypts A and B using K_S^S to obtain n_1 and $(IDS \oplus n_1)$ and extracts IDS. Given IDS, the server can retrieve tag's private key and shared secret keys. Then S decrypts C to authenticate T_i. Assuming T_i is authenticated, S computes $D = \{(n_1 \oplus K_m)^2 \ mod \ K_P^T\} \oplus K_l$ and sends D to T_i. S also updates the tag's group index and secrets.
5) T_i receives D and verifies it to authenticate R and S. Then T_i also updates its group index and secrets, assuming the authentication was successful.
6) In the updating step for T_i, it updates its secret key K_x, for $x \in \{h, m, l\}$, as $K_x = K_x \oplus n_1$ and updates its IDS as $IDS = IDS \oplus n_1$.
7) In the updating step for S, it first stores the current set of the keys and IDS which has been used to authenticate the tag as *old* set of the information for T_i, i.e. $(K_h^{old}, K_m^{old}, K_l^{old}, IDS)$, then it calculates secret key K_x^{new}, for $x \in \{h, m, l\}$, as $K_x^{new} = K_x \oplus n_1$ and new IDS as $IDS^{new} = IDS^{new} \oplus n_1$.

III. SECURITY ANALYSIS OF Gao *et al.* PROTOCOLS

In this section, we analyse the security of the mutual authentication protocols that have been proposed by Gao *et al.* in more details.

A. SECURITY ANALYSIS OF HASH BASED DYNAMIC GROUP-INDEXING AUTHENTICATION PROTOCOL

Here, we analyse the security of the hash based dynamic grouping authentication protocol which has been proposed by Gao *et al.* [13] and highlight its important weaknesses.

Despite of the designers' claim, it is easy to desynchronize the tag from the server in this protocol and rather trivial. More precisely, the last message is sent by T_i while R/S keeps the history of the old parameters to avoid desynchronization attack, based on the designers' claim [13, Sec. IV.C.,P. 8380]. However, if the adversary intercepts the message which is sent from T_i to R/S then T_i has updated its secrets while R/S has not. Hence they will be desynchronized. In more details, assume that the current history of the T_i are K_h, K_m, K_l, g_i and the R/S records for T_i are also K_h, K_m, K_l, g_i as the new records and K_h^{old}, K_m^{old}, K_l^{old}, g_i^{old} as the old records. Next, the adversary involves in a session of the protocol between T_i and R/S as follows:

1) R/S generates a random number n_1 and sends it to T_i.
2) T_i replies with $A = h(g_i)$ and $B = h(K_h \oplus g_i \oplus n_1)$.
3) Given $A = h(g_i)$, R/S finds the group index of T_i among the tags in this group, based on the received $B = h(K_h \oplus g_i \oplus n_1)$. So, T_i is authenticated by R/S.

Then R/S generates a random number n_2, calculates $C = h(K_m \oplus n_2)$ and sends them to T_i.

4) T_i evaluates the received message C and authenticates R/S. Then T_i generates a random number n_3, calculates $D = h(K_l \oplus n_2) \oplus n_3$ and sends them to R/S. T_i also updates its group index and secrets as $K_x' = h(K_x \oplus n_2 \oplus n_3)$, for $x \in \{h, m, l\}$ and updates its grouping index to g_j, where $j = n_3 \ mod \ m$, and m is the number of groups in the database of the reader.
5) The adversary blocks D. Hence the server does not update its secret for T_i.

After the above procedure of attack, T_i records are K_h', K_m', K_l' and g_j, where $K_x' = h(K_x \oplus n_2 \oplus n_3)$, for $x \in \{h, m, l\}$. On the other hand, the R/S records for T_i are K_h, K_m, K_l, g_i as the new records and K_h^{old}, K_m^{old}, K_l^{old}, g_i^{old} as the old records. It is clear that the T_i records match none of the R/S records with a high probability. Hence T_i and R/S have been desynchronized. The complexity of the given attack is just a session of the protocol while the success probability of the attack is $(1 - 2^{-m} \times 2^{-3\times\ell})^2 \approx 1$, where ℓ is output length of the hash function and m is number of grouping indexes.

It should be noted this protocol also has another weakness in which the server is not able to verify the correctness of the received D, even if the tag sends n_3 in the plain form. More precisely, assume that in the last step, the tag sends $D = h(K_l \oplus n_2) \oplus n_3$ and n_3 to R/S. Now the adversary can intercept it and send $D' = D \oplus \Delta$ and $n_3' = n_3 \oplus \Delta$ instead, to R/S. Now, R/S authenticates the received message and updates the tag's secrets based on $n_3 \oplus \Delta$ while T_i has updated them based on n_3. Hence, again T_i and R/S will be desynchronized. The complexity of this attack is also just a session of the protocol while the success probability of the attack is $(1 - 2^{-m} \times 2^{-3\times\ell})^2 \approx 1$.

It worth noting, as long as the tag has not updated its secrets, it is possible to trace it. More precisely, given T_i, if the adversary initiates a session by sending an arbitrary n_1, the tag will reply with $A = h(g_i)$ and $B = h(K_h \oplus g_i \oplus n_1)$. While $A = h(g_i)$ is identical for all tags in the list $\mathcal{L}_i$, $B = h(K_h \oplus g_i \oplus n_1)$ could be unique for T_i, given that it is a function of its secret key K_h. Hence, given a tag T_i, the adversary can first desynchronize it, based on the given attack and then use the given property to trace it. Hence, this protocol also suffers from a traceability attack.

B. SECURITY ANALYSIS OF RABIN BASED AUTHENTICATION PROTOCOL

Here, we analyse the security of the Rabin public key authentication protocol which has been proposed by Gao *et al.* [13] and highlight its important weaknesses.

Similar to the hash based protocol and again despite the designers' claim, it is easy to desynchronize the tag from the server in this protocol as well. The source of the problem in this protocol is the fact that only the tag contributes to the randomness of the protocol. Hence, based on the Safkhani and Bagheri attack on generalized authentication protocols [15], it is always possible to desynchronize this type of protocols. Assuming that the current records of the tag in the reader's side are $(K_h^{old}, K_m^{old}, K_l^{old}, IDS^{old}, K_h, K_m, K_l, IDS, K_S^T, K_P^T, K_S^S, K_P^S)$ and its records in the tag's side are $(K_h, K_m, K_l, IDS, K_S^T, K_P^T, K_P^S)$, the attack procedure is as follows:

1) **In session i:**
Assuming the legitimate reader R communicates with the legitimate tag T_i,
   a) R sends *Hello* and receives (A^i, B^i, C^i) from T_i, where $A^i = n_1^2 \ mod \ K_P^S$, $B^i = (IDS \oplus n_1)^2 \ mod \ K_P^S$, $C^i = m_1^2 \ mod \ K_P^T$ and $m_1 = n_1 \oplus K_h$.
   b) R forwards the received (A^i, B^i, C^i) to the server S, S authenticates T_i and computes $D^i = \{(n_1 \oplus K_m)^2 \ mod \ K_P^T\} \oplus K_l$ and sends D^i to T_i.
   c) The adversary $\mathcal{A}$ eavesdrops A^i, B^i, C^i and D^i and stores them. Furthermore, $\mathcal{A}$ blocks D^i. Hence, the tag's record remains unchanged while S updates them. Hence, the server records in this stage are $(K_h, K_m, K_l, IDS, K_h^{new}, K_m^{new}, K_l^{new}, IDS^{new}, K_S^T, K_P^T, K_S^S, K_P^S)$, where $K_x^{new} = K_x \oplus n_1$, for $x \in \{h, m, l\}$ and $IDS^{new} = IDS \oplus n_1$. It is clear the records in the tag's side are $(K_h, K_m, K_l, IDS, K_S^T, K_P^T, K_P^S)$.
2) In session $i + 1$:
   a) The reader R sends *Hello* and receives $(A^{i+1}, B^{i+1}, C^{i+1})$ from T_i, where $A^{i+1} = n_1'^2 \ mod \ K_P^S$, $B^{i+1} = (IDS \oplus n_1')^2 \ mod \ K_P^S$, $C^{i+1} = m_1'^2 \ mod \ K_P^T$ and $m_1' = n_1 \oplus K_h$.
   b) R forwards the received $(A^{i+1}, B^{i+1}, C^{i+1})$ to the server S, S authenticates T based on its old records, computes $D^{i+1} = \{(n_1' \oplus K_m)^2 \ mod \ K_P^T\} \oplus K_l$ and sends D^{i+1} to T_i and updates the records.
   c) T_i authenticates S and updates its records.
   d) Up to this stage, the tag's records are K_h^{new1}, K_m^{new1}, K_l^{new1}, IDS^{new1}, K_S^T, K_P^T, K_P^S), where $K_x^{new1} = K_x \oplus n_1'$, for $x \in \{h, m, l\}$ and $IDS^{new1} = IDS \oplus n_1'$. The server records in this stage are $(K_h, K_m, K_l, IDS, K_h^{new1}, K_m^{new1}, K_l^{new1}, IDS^{new1}, K_S^T, K_P^T, K_S^S, K_P^S)$.
3) In session $i + 2$:
   a) R sends *Hello*.
   b) The adversary impersonates the tag and replies with the stored values of (A^i, B^i, C^i), where $A^i = n_1^2 \ mod \ K_P^S$, $B^i = (IDS \oplus n_1)^2 \ mod \ K_P^S$, $C^i = m_1^2 \ mod \ K_P^T$ and $m_1 = n_1 \oplus K_h$.
   c) R forwards the received (A^i, B^i, C^i) to the server S, S authenticates T and computes $D^i = \{(n_1 \oplus K_m)^2 \ mod \ K_P^T\} \oplus K_l$ and sends D^i to T_i.
   d) S also updates its records in this stage to $(K_h, K_m, K_l, IDS, K_h^{new}, K_m^{new}, K_l^{new}, IDS^{new}, K_S^T, K_P^T, K_S^S, K_P^S)$, where

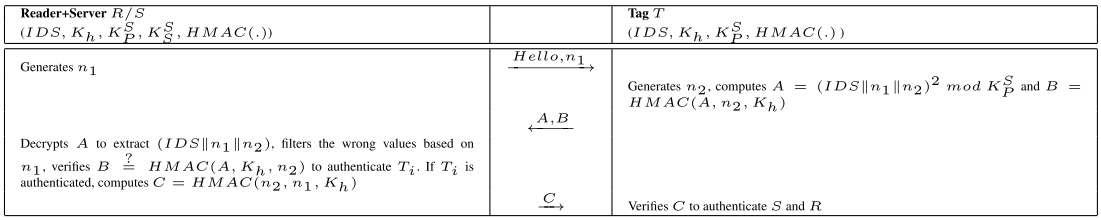


FIGURE 3. The enhanced protocol based on Rabin public key and one-way hash function.

$K_x^{new} = K_x \oplus n_1$, for $x \in \{h, m, l\}$ and $IDS^{new} = IDS \oplus n_1$.

e) Given that the tag's records are K_h^{new1}, K_m^{new1}, K_l^{new1}, $IDS^{new1}, K_S^T, K_P^T, K_P^S$), where $K_x^{new1} = K_x \oplus n_1'$, for $x \in \{h, m, l\}$ and $IDS^{new1} = IDS \oplus n_1'$, the tag and the server have been desynchronized.

Following this attack, we can desynchronize the tag and the server with the complexity of only three sessions of the protocol and the success probability of $1 - 2^{-r}$, where r is the bit length of the random number generated by the tag.

It should be noted, after desynchronization of T_i from the server, the tag's records on the server's side remain unchanged. Hence, at any time, when the server sends a *Hello* request, the adversary can reply with the stored (A^i, B^i, C^i) to impersonate the tag. Given that S includes $(K_h, K_m, K_l, IDS, K_h^{new}, K_m^{new}, K_l^{new}, IDS^{new}, K_S^T, K_P^T, K_S^S, K_P^S)$ for T_i and the adversary knows $A^{i+1} = n_1'^2 \ mod \ K_P^S$, $B^{i+1} = (IDS \oplus n_1')^2 \ mod \ K_P^S$, $C^{i+1} = m_1'^2 \ mod \ K_P^T$ and $m_1' = n_1 \oplus K_h$, the adversary will be successfully authenticated by the server at any desired time, assuming the server has sent a *Hello* request. Assuming that the adversary has already successfully desynchronized the tag, the success probability of the tag impersonation attack is one and the complexity is just a session of the protocol.

IV. ENHANCED RABIN BASED AUTHENTICATION PROTOCOL

In this section, we propose an amended version of the Rabin based protocol by Gao *et al.*. We assume that in the registration phase of the protocol, T_i and S have shared K_h and IDS respectively as the secret key and the identifier of the tag. In addition, the public key of the server, i.e. K_P^S, is known by any tag. Moreover, we assume that T_i also supports a one-way hash function based MAC such as $HMAC(.)$. As it is depicted in Figure 3, the enhanced protocol process runs as follows between R, S and T_i:

1) R generates a random number n_1 and sends it along a *Hello* request to T_i.
2) T_i generates a random number n_2, computes $A = (IDS\|n_1\|n_2)^2 \ mod \ K_P^S$ and $B = H(A, n_2, K_h)$ and sends (A, B) to R.
3) R forwards the received values to the server S.
4) S decrypts A to extract real value of $(IDS^*\|n_1^*\|n_2^*)$, using n_1^*, verifies $B \stackrel{?}{=} H(A, K_h, n_2^*)$ to authenticate T_i. Assuming T_i is authenticated, S computes $C = H(n_2, n_1, K_h)$ and sends C to T_i.
5) T_i receives C and verifies it to authenticate R and S.

V. SECURITY AND PERFORMANCE ANALYSIS OF THE PROPOSED PROTOCOL

To evaluate the security of the proposed protocol, in this section we investigate its security informally and also formally in Real-or-Random model (RoR). We also provide comparisons between the enhanced protocol and the Gao *et al.* protocol from different perspectives.

A. INFORMAL SECURITY ANALYSIS

In order to assess the resistance of the improved protocol based on the informal method, we present an adversary model with some assumptions. We consider that the adversary has access to communication channels and can eavesdrop all transferred messages. She/he can intercept the line and transfer her/his packets to the tag or the reader. In addition, she/he is able to run all functions–such as PRNG and encryption–without having access to secret keys.

1) TRACEABILITY ATTACK

To perform the traceability attack on a protocol, the adversary should be able to connect the transferred messages over different sessions or link them with the protocol's party. In the enhanced protocol, the transferred messages over public channel are n_1, $A = (IDS\|n_1\|n_2)^2 \ mod \ K_P^S$, $B = HMAC(A, n_2, K_h)$ and $C = HMAC(n_2, n_1, K_h)$, where n_1 and n_2 are random values that are respectively contributed by the reader and the tag. Assuming that K_P^S has been selected properly and the $HMAC(.)$ is a secure one-way hash function based MAC, then whole transferred messages over each session will be random to $\mathcal{A}$ and she/he can not distinguish them from random sequences in polynomial time. Hence, the enhanced protocol is resistant against the traceability attack.

2) SECRET DISCLOSURE ATTACK

The secret disclosure attack occurs when $\mathcal{A}$ can extract the tag's or the reader's confidential information (e.g. encryption keys or identification values). In the enhanced protocol, n_1 does not include any secret parameter, B and C are protected by $HMAC(.)$ and extracting any information from A is equivalent to dealing with the factoring problem. Therefore,

the proposed protocol is secure against secret disclosure attack.

3) REPLAY ATTACK

Given that, in the enhanced protocol, both the tag and the reader are contributing to the randomness of the sensitive transferred messages, the adversary can not replay the eavesdropped messages from an old session in a later one. Hence, the improved protocol is resistant against the replay attack.

4) IMPERSONATION ATTACK

To impersonate a protocol party, $\mathcal{A}$ should provide a valid response to the given challenge by the other party. More precisely, to impersonate T_i, given the fresh value n_1, the adversary should generate a valid pair $A = (IDS\|n_1\|n_2)^2 \ mod \ K_P^S$ and $B = HMAC(A, n_2, K_h)$, without the knowledge of IDS and K_h. Hence, the adversary's advantage to impersonate the tag is negligible. Similarly, to impersonate the reader/server, $\mathcal{A}$ should provide a valid $C = HMAC(n_2, n_1, K_h)$, without the knowledge of n_2 and K_h. Even if we release n_2, the adversary yet can not impersonate the reader without the knowledge of K_h in polynomial time. Hence, the enhanced protocol is resistant against the impersonation attack.

5) DESYNCHRONIZATION ATTACK

If the tag or the reader update a shared value, the adversary may could desynchronize them by forcing them to keep unmatched values, similar to the proposed attacks against the Gao *et al.*'s protocols. However, in the proposed protocol, the protocol's parties do not update any value. Hence, it is secure against desynchronization attack.

B. FORMAL ANALYSIS

In subsection V-A, we analysed the security of the enhanced protocol against various attacks heuristically. In this subsection, we use formal approaches to validate its robustness. To evaluate the security of a cryptographic protocol formally, different approaches are possible including theoretical models such as Real-or-Random (RoR) and Find-then-Guess, manual logic-based models such as GNY logic [16] and BAN logic [17] and automatic on-the-shelves tools such as Scyther [18], AVISPA [19], Proverif [20] and CryptoVerif [21]. We use Real-or-Random model and Scyther tool to evaluate the security of the proposed protocol formally, that are two widely accepted approaches to evaluate the security of a cryptographic protocol.

C. FORMAL SECURITY ANALYSIS OF THE REVISED PROTOCOL IN RoR MODEL

In this section, following [22], we formally evaluate the security of the enhanced protocol in Real-or-Random model (RoR), by determining the adversary's advantage in distinguishing the real world of the enhanced protocol from the random world (RW), for simplicity we denote the enhanced protocol by EP.

Theorem 1: The adversary's advantage to distinguish EP from RW after respectively q_{exe}, q_{send} and q_{test} queries to Execute, Send and Test oracles on EP/RW, is:

$$\begin{aligned} &Adv^{RoR}_{\mathcal{D},RP}(t, q_{exe}; q_{test}; q_{send}) \\ &\quad - Adv^{RoR}_{\mathcal{D},RW}(t, q_{exe}; q_{test}; q_{send}) \\ &\leq q.\varepsilon_{Fact} + 2.q.\varepsilon_{HMAC} \end{aligned}$$

where, we assume at least $l = |n_1| = |n_2|$ bits of the input of each function is random, ε_{Fact} denotes the maximum advantage of solving the factoring problem on each query and ε_{HMAC} denotes the maximum advantage of contradicting collision resistance property of $HMAC(.)$ and $q = q_{exe} + q_{test} + q_{send}$.

Proof: Let the tag T_i and the reader R_j are communicating to share a session key and let $\mathcal{A}$ be an adversary against the semantic security of EP in the RoR model, given that the channel between the reader and the server is secure, we consider both as the reader R. To prove the Theorem 1, we are using a game based approach and defining a series of games $\mathcal{G}$, starting from random world RW and ended in real world EP. For each game $\mathcal{G}_n$, we define an event $Adv^{RoR-\mathcal{G}_n}_{\mathcal{D},\mathcal{P}}(t, R)$ corresponding to the adversary's advantage to correctly guess the hidden bit b involved in the Test queries. This advantage is used to determine the adversary's gains while switching from one game to another.

Game $\mathcal{G}_0$.: It defines RW and any transferred message is selected uniformly random from related domain and $Adv^{RoR-\mathcal{G}0}_{\mathcal{D},RW}(t, R) = 0$.

Game $\mathcal{G}_1$.: Compared to $\mathcal{G}_0$, in this game we use the real value of $A = (IDS\|n_1\|n_2)^2 \ mod \ K_P^S$, where n_1 and n_2 are fresh random values on each session and $\mathcal{A}$ can not control both. Hence, we can assume A is computed as $(IDS\|r)^2 \ mod \ K_P^S$, where at least half of bits of r are random and it leads to an unpredictable result for A. Hence:

$$Adv^{RoR-\mathcal{G}0}_{\mathcal{D},RW}(t, R) - Adv^{RoR-\mathcal{G}1}_{\mathcal{D},RW}(t, R) \leq q.\varepsilon_{Fact}$$

where $q = q_{exe} + q_{test} + q_{send}$.

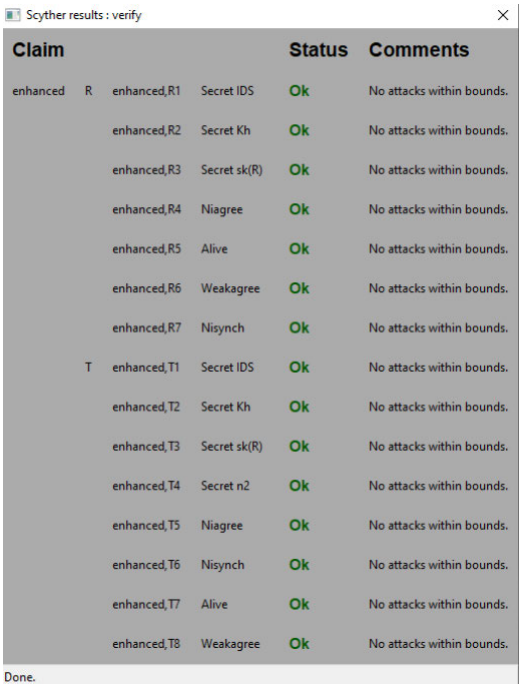
Game $\mathcal{G}_2$.: This game is identical to G_1 with an exception that $B = HMAC(A, n_2, K_h)$, where K_h is a secure parameter and A is a randomized value following $\mathcal{G}_1$. Given that the output of a secure MAC such as HMAC is not distinguishable from a random oracle up to collision resistant bound, we can conclude that:

$$Adv^{RoR-\mathcal{G}_2}_{\mathcal{D},RW}(t, R) - Adv^{RoR-\mathcal{G}_1}_{\mathcal{D},RW}(t, R) \leq q.\varepsilon_{HMAC}.$$

Game $\mathcal{G}_3$.: In this game, we use $C = HMAC(n_2, n_1, K_h)$, where again K_h is a secure parameter and $\mathcal{A}$ can not control both n_1 and n_2. Hence, C is undistinguishable from a random value as long as $HMAC(.)$ is undistinguishable. Therefore:

$$Adv^{RoR-\mathcal{G}_3}_{\mathcal{D},RW}(t, R) - Adv^{RoR-\mathcal{G}_2}_{\mathcal{D},RW}(t, R) \leq q.\varepsilon_{HMAC}.$$

Game $\mathcal{G}_4$.: This game is identical to EP because $A = (IDS\|n_1\|n_2)^2 \ mod \ K_P^S$, $B = HMAC(A, n_2, K_h)$ and



(a) Verification results

```
const SQMOUDULO:Function;
const Con:Function;
hashfunction HMAC;

secret  Kh;
secret IDS;
const Hello;

macro A=SQMOUDULO(Con(IDS,n1,n2),pk(R));
macro B=HMAC(A,Kh,n2);
macro C=HMAC(n2,n1,Kh);

protocol enhanced( R, T)
{
    role R
    {
         fresh n1;
        var n2;
        secret  Kh; secret IDS;
        send_1(R,T,Hello,n1);
        recv_2(T,R,A,B);
        send_3(R,T,C);
        claim(R, Secret, IDS);  claim(R, Secret, Kh);
        claim(R, Secret,sk(R)); claim(R,Niagree);
        claim(R,Alive);  claim(R,Weakagree);
        claim(R,Nisynch);
    };
    role T
    {
        fresh n2;
        ar n1;
        secret  Kh;     secret IDS;
        recv_1(R,T,Hello,n1);
        send_2(T,R,A,B);
        recv_3(R,T,C);
        claim(T, Secret,IDS);
        claim(T, Secret, Kh);  claim(T, Secret, sk(R));
        claim(T, Secret, n2);  claim(T,Niagree);
        claim(T,Nisynch); claim(T,Alive);
        claim(T,Weakagree);
};
}
```

(b) Enhanced protocol in SPDL

FIGURE 4. The Scyther tool's report for the security verification of the enhanced protocol.

$C = HMAC(n_2, n_1, K_h)$. On the other hand, the transferred messages are identical to those in $\mathcal{G}_3$. Hence:

$$\begin{aligned} &Adv^{RoR}_{\mathcal{D},EP}(t,R) - Adv^{RoR}_{\mathcal{D},RW}(t,R) \\ &\quad \leq Adv^{RoR-\mathcal{G}_4}_{\mathcal{D},RW}(t,R) - Adv^{RoR-\mathcal{G}_0}_{\mathcal{D},RW}(t,R) \\ &\quad \leq q.\varepsilon_{Fact} + 2.q.\varepsilon_{HMAC} \end{aligned}$$

which completes the proof. ■

1) FORMAL SECURITY ANALYSIS USING SCYTHER TOOL

We also use the Scyther tool [18] to validate the security of the enhanced protocol formally. Scyther is a widely accepted tool to verify the security of cryptographic protocols automatically, which follows Dolev-Yao (DY) adversary model [23]. To evaluate the security of a protocol using this tool, the protocol should be described using the Security Protocol Description Language (SPDL) and analyzed by the Scyther tool. Figure 4 represents SPDL modeling of the enhanced protocol and its verification results using the Scyther tool, which confirms its security.

D. SECURITY COMPARISON

In Table 2, we compare the security of the enhanced protocol with the Gao *et al.*'s protocol. While both protocol provide security against secret disclosure attack, and replay attack, the proposed protocol also provides security against other attacks as well.

TABLE 2. Security comparison, where A_1, A_2, A_3, A_4 and A_5 respectively denote security against replay, secret disclosure, impersonation, traceability and desynchronization attacks.

Protocol	A_1	A_2	A_3	A_4	A_5
Gao *et al.*	✓	✓	×	×	×
Enhanced	✓	✓	✓	✓	✓

E. COMPUTATION COMPARISON

To achieve 80-bit security, we consider the output of HMAC to be 160 bits and $|p \times q| = 1024$. Given an string x, the required time for calculation modulo squaring operation $x^2 \ mod \ p \times q$ is denoted by T_{MS}, given $x^2 \ mod \ p \times q$, the required time for squaring root solving operation and finding x is denoted by T_{SR}, the required time for calculating $HMAC(x)/H(x)$ is denoted by T_H and the required time to generate a random number is indicated by T_R. Following this assumption, the computation overhed of a tag of Gao *et al.* protocol is at least $4 \times T_{MS} + T_R$ while the computation cost of a tag in the enhanced protocol is $T_{MS} + 2 \times T_H + T_R$. The server/reader of Gao *et al.*'s protocol cost is at least $3 \times T_{SR} + T_{MS}$ while the computational cost of the server/reader of the enhanced protocol is $T_{SR} + 2 \times T_H + R_R$, in the same setting. Table 3 provides a computational comparison of the two protocols with two other related works [24], [25].

F. COMMUNICATION COMPARISON

Based on the parameter setting of subsection V-E, the length of each transferred parameter in Gao *et al.* protocol, i.e.

TABLE 3. Computation comparison.

Protocol	T_i-cost	R/S-cost
Ref. [24]	$3 \times T_{MS} + T_R + 2 \times T_H$	$3 \times T_{MS} + 6 \times T_{SR} + T_R + 11 \times T_H$
Ref. [25]	$3 \times T_{MS} + 3 \times T_R$	$3 \times T_{MS} + 4 \times T_{SR} + 3 \times T_R + T_H$
Ref. [26]	$T_{MS} + 2 \times T_{SR} + T_R$	$4 \times T_{SR} + 3 \times T_R$
Gao *et al.*	$4 \times T_{MS} + T_R$	$3 \times T_{SR} + T_{MS}$
Enhanced	$T_{MS} + T_R + 2 \times T_H$	$T_{SR} + T_R + 2 \times T_H$

TABLE 4. Computation comparison, where T_i-R, T_i-T, R/S-R and R/S-T respectively denote the total bits received by the tag, transferred by the tag, received by the reader/server and transferred by the reader/server.

Protocol	T_i-R	T_i-T	R/S-R	R/S-T
Ref. [24]	480	3072	3072	480
Ref. [25]	416	2048	2048	416
Ref. [26]	1088	2368	2368	1088
Gao *et al.*	1024	3072	3072	1024
Enhanced	1280	1184	1184	1280

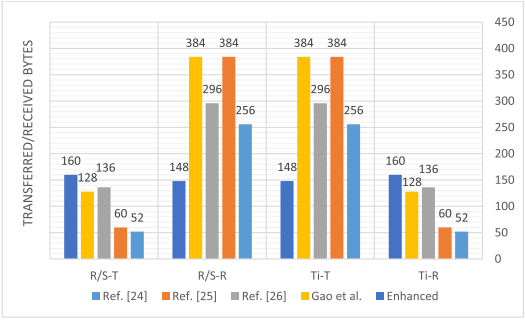


FIGURE 5. Enhanced protocol versus related protocols, communications comparison.

A, B, C and D, is 1025 bits. Hence, in this protocol, T_i transfers 3072 bits and receives 1024 bits. In the enhanced protocol, to make sure $x^2 > K_P^S$ we consider the length of any generated random number 256 bits. In this setting, the T_i receives $256 + 1024$ bits and transfers $160 + 1024$ bits. Similarly, S/R receives $160 + 1024$ bits and transfers $256 + 1024$ bits. Table 4 provides a detailed comparison of the two protocols in terms of communicated bits and its graphical representation is provided in Figure 5. The length of timestamps assumed to be 64 bits. In addition, we omitted the communications costs between the server and the reader.

VI. CONCLUSIONS AND DISCUSSION

In this paper, we analysed the security of improved protocols that have recently proposed by Gao *et al.* Although the designers claimed that those protocols provide strong security against known attacks in the context, however, we presented efficient approaches to desynchronize those protocols and also efficiently extend the proposed desynchronization attacks to either the traceability attack, in the case hash based protocol or to the tag impersonation attack, in the case of Rabin public key cryptography based protocol. Although it is common in the literature to break an ultralightweight mutual authentication protocol with just a few queries and with a high probability, however, the analysed protocols in this paper are not ultralightweight and they used strong cryptographic components as the source of their security, i.e. hash function and public key cryptography. The main reason for the success of the proposed attacks is the weak structure of the transferred messages in which the adversary capable to compromise the protocols efficiently.

This study, behind many interesting related literature, shows that designing a secure protocol is not a straightforward task. In addition, it once again shows that to provide desired security it is not enough to just use a secure component and all details of the protocol are important.

Some basic recommendations for the designers of an authentication protocol are as follows:

- Any protocol's session should be randomized by fresh nonces and all the protocol's parties should have contribute to the randomization.
- Any sensitive data which is transferred over an insecure channel should be properly encrypted.
- The integrity of all sensitive messages should be guaranteed and the adversary should not be able to manipulate a message without been detected.
- It should not be possible to link different messages from different sessions.
- If the party $\mathcal{X}$ evaluates the message $\mathcal{M}$ to authenticate the party $\mathcal{Y}$, then $\mathcal{X}$ should had already contributed to the randomness of $\mathcal{M}$.

If a designer considers those recommendations in designing a protocol, the designed protocol will be secure against many common attacks on protocols.

Following the given recommendations, we proposed an enhanced version of the Rabin based protocol of Gao *et al.* and evaluated its security against various attacks formally and informally. The security analysis demonstrated that the enhanced protocol provides desired security against different attacks such as traceability, impersonation and desynchronization attacks.

ACKNOWLEDGMENT

The authors gratefully thank all the anonymous reviewers for their valuable comments, which helped them to improve the technical content and presentation of the work significantly. They would also like to thank Masoumeh Safkhani for her invaluable helps and discussions through this work. The article was created with using institutional support for long-term conceptual development of research organization of the University of Finance and Administration.

REFERENCES

[1] M. A. Ferrag, L. A. Maglaras, H. Janicke, J. Jiang, and L. Shu, "Authentication protocols for Internet of Things: A comprehensive survey," *Secur. Commun. Netw.*, vol. 2017, Oct. 2017, Art. no. 6562953.
[2] I. Markit. *Number of Connected IoT Devices Will Surge to 125 Billion by 2030*. Accessed: Mar. 29, 2020. [Online]. Available: https://technology.ihs.com/596542
[3] I. Markit, *The Internet of Things: A Movement, Not a Market*, vol. 28. Englewood, CO, USA: IHS Markit, 2017. Accessed: Mar. 23, 2020. [Online]. Available: https://cdn.ihs.com/www/pdf/IoT_ebook.pdf
[4] V. Rozsa, M. Denisczwicz, M. L. Dutra, P. Ghodous, C. F. da Silva, N. Moayeri, F. Biennier, and N. Figay, "An application domain-based taxonomy for IoT sensors," in *Transdisciplinary Engineering: Crossing Boundaries* (Advances in Transdisciplinary Engineering), vol. 4, M. Borsato, N. Wognum, M. Peruzzini, J. Stjepandic, and W. J. C. Verhagen, Eds. Parana, Brazil: IOS Press, Oct. 2016, pp. 249–258.
[5] A. I. A. Ahmed, A. Gani, S. H. A. Hamid, A. Abdelmaboud, H. J. Syed, R. A. A. Habeeb Mohamed, and I. Ali, "Service management for IoT: Requirements, taxonomy, recent advances and open research challenges," *IEEE Access*, vol. 7, pp. 155472–155488, 2019.
[6] S. A. Shah, D. Z. Seker, S. Hameed, and D. Draheim, "The rising role of big data analytics and IoT in disaster management: Recent advances, taxonomy and prospects," *IEEE Access*, vol. 7, pp. 54595–54614, 2019.
[7] M. Presser and A. Gluhak, *The Internet of Things: Connecting the Real World With the Digital World*, vol. 2. EURESCOM Mess ge–The Magazine for Telecom Insiders, 2009.
[8] R. Baashirah and A. Abuzneid, "Survey on prominent RFID authentication protocols for passive tags," *Sensors*, vol. 18, no. 10, p. 3584, Oct. 2018.
[9] NIST, Gaithersburg, MD, USA. (2018). *Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process*. [Online]. Available: csrc.nist.gov/CSRC/media/Projects/Lightweight-Cryptography/documents/final-lwc-submission-requirements-august2018.pdf
[10] M. Safkhani, "On the security of Tan serverless RFID authentication and search protocols," in *Proc. 8th Int. Workshop* (Lecture Notes in Computer Science), vol. 7739, J. Hoepman and I. Verbauwhede, Eds. Nijmegen, The Netherlands: Springer, Jul. 2012, pp. 1–19.
[11] B. Wang and M. Ma, "A server independent authentication scheme for RFID systems," *IEEE Trans. Ind. Informat.*, vol. 8, no. 3, pp. 689–696, Aug. 2012.
[12] C. Tan, B. Sheng, and Q. Li, "Secure and serverless RFID authentication and search protocols," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1400–1407, Apr. 2008.
[13] L. Gao, L. Zhang, F. Lin, and M. Ma, "Secure RFID authentication schemes based on security analysis and improvements of the USI protocol," *IEEE Access*, vol. 7, pp. 8376–8384, 2019.
[14] S. Galbraith, "The RSA and Rabin cryptosystems," Dept. Math., Cyber Secur. Foundry, Univ. Auckland, Auckland, New Zealand, Tech. Rep., 2012.
[15] M. Safkhani and N. Bagheri, "Generalized desynchronization attack on UMAP: Application to RCIA, KMAP, SLAP and SASI$^+$ protocols," Cryptol. ePrint Arch., Tech. Rep. 2016/905, 2016. [Online]. Available: https://eprint.iacr.org/2016/905
[16] L. Gong, R. Needham, and R. Yahalom, "Reasoning about belief in cryptographic protocols," in *Proc. IEEE Comput. Soc. Symp. Res. Secur. Privacy*, Oct. 1990, pp. 234–248.
[17] M. Burrows, M. Abadi, and R. Needham, "BAN: A logic of authentication," Digit. Equip. Syst. Res. Center, Palo Alto, CA, USA, Tech. Rep. 39, 1989.
[18] C. J. F. Cremers, "The Scyther tool: Verification, falsification, and analysis of security protocols," in *Computer Aided Verification*. Berlin, Germany: Springer, 2008, pp. 414–418.
[19] A. Armando, D. A. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron, "The AVISPA tool for the automated validation of internet security protocols and applications," in *Proc. 17th Int. Conf. Comput. Aided Verification (CAV)*, in Lecture Notes in Computer Science, vol. 3576, K. Etessami and S. K. Rajamani, Eds. Edinburgh, U.K.: Springer, Jul. 2005, pp. 281–285.
[20] B. Blanchet and A. Chaudhuri, "Automated formal analysis of a protocol for secure file sharing on untrusted storage," in *Proc. IEEE Symp. Secur. Privacy*, May 2008, pp. 417–431.
[21] B. Blanchet, "A computationally sound mechanized prover for security protocols," in *Proc. IEEE Symp. Secur. Privacy*, 2006, p. 117.
[22] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Public Key Cryptography*, vol. 3386. Berlin, Germany: Springer, 2005, pp. 65–84.
[23] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.
[24] Z. Zhou, P. Wang, and Z. Li, "A quadratic residue-based RFID authentication protocol with enhanced security for TMIS," *J. Ambient Intell. Hum. Comput.*, vol. 10, no. 9, pp. 3603–3615, Sep. 2019.
[25] R. Doss, S. Sundaresan, and W. Zhou, "A practical quadratic residues based scheme for authentication and privacy in mobile RFID systems," *Ad Hoc Netw.*, vol. 11, no. 1, pp. 383–396, Jan. 2013.
[26] J. Zhou, "A quadratic residue-based lightweight RFID mutual authentication protocol with constant-time identification," *J. Commun.*, vol. 10, no. 2, pp. 117–123, 2015.

MEHDI HOSSEINZADEH received the B.S. degree in computer hardware engineering from Islamic Azad University, Dezfol Branch, Iran, in 2003, and the M.Sc. and Ph.D. degrees in computer system architecture from the Science and Research Branch, Islamic Azad University, Tehran, Iran, in 2005 and 2008, respectively. He is currently an Associate Professor at the Iran University of Medical Sciences (IUMS), Tehran. He is the author/coauthor of more than 120 publications in technical journals and conferences. His research interests include SDN, information technology, data mining, big data analytics, e-commerce, e-marketing, and social networks.

OMED HASSAN AHMED (Member, IEEE) received the B.Sc. and H.N.D. (Higher National Diploma) degrees in information technology from Teesside University, U.K., and the M.Sc. degree in computer science from Newcastle University, U.K. He is currently a Ph.D. Researcher at Huddersfield University, U.K. He is also the Head of the Information Technology Department, College of Science and Technology, University of Human Development, Iraq, where he has been a Faculty Member, since 2013.

SARKAR HASAN AHMED received the bachelor's degree from Duhok University, Iraq, in 2009, and the master's degree in software development from Coventry University, U.K., in 2013. He is currently pursuing the Ph.D. degree in the field of IoT and security with Sulaimani Polytechnic University. He is also a Researcher and a Lecturer at Sulaimani Polytechnic University, Sulaymaniyah, Iraq.

CUONG TRINH received the M.Sc. degree in computer science from Military Technical Academy, Vietnam, in 2014. He is currently pursuing the Ph.D. degree in informatics, communication technology, and applied mathematics with the Technical University of Ostrava, Czech Republic. His research interests include data mining, parallel computing, network infrastructure, and network security. He has a lot of experience in system network deployment, implemented RFID applications for some fields, such as a library, education, transportation system.

NASOUR BAGHERI received the M.S. and Ph.D. degrees in electrical engineering from the Iran University of Science and Technology (IUST), Tehran, Iran, in 2002 and 2010, respectively. He is currently an Associate Professor at the Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran. He is also a part-time Researcher with the Institute for Research in Fundamental Sciences. He is the author of more than 100 articles in information security and cryptology. His research interests include cryptology, more precisely, designing and analysis of symmetric schemes, such as lightweight ciphers, e.g., block ciphers, hash functions, and authenticated encryption schemes, cryptographic protocols for constrained environment, such as RFID tags and the IoT edge devices and hardware security, e.g., the security of symmetric schemes against side-channel attacks, such as fault injection and power analysis.

SARU KUMARI received the Ph.D. degree in mathematics from Chaudhary Charan Singh University, Meerut, India, in 2012. She is currently an Assistant Professor at the Department of Mathematics, Chaudhary Charan Singh University. She has published more than 133 research articles in reputed International journals and conferences, including 115 publications in SCI-indexed journals. Her current research interests include information security and applied cryptography. She is a Technical Program Committee member for many International conferences. She has served as a Lead/Guest Editor for four special issues in SCI journals of Elsevier, Springer, and Wiley. She is on the Editorial Board of more than 12 journals of international repute, including seven SCI journals.

JAN LANSKY received the M.S. and Ph.D. degrees in computer science (software systems) from Charles University, Prague, Czech Republic, in 2005 and 2009, respectively. He has been a Professor with the Department of Computer Science and Mathematics, Faculty of Economic Studies, University of Finance and Administration, Prague, since March 2009, where he has been the Head of the Department, since September 2014. His research interests include cryptocurrencies, text compression, and databases.

BAO HUYNH received the M.Sc. degree in computer science from the Posts and Telecommunications Institute of Technology, Ho Chi Minh City, Vietnam, in 2011, and the Ph.D. degree in computer science from the VSB—Technical University of Ostrava, Czech Republic, in 2017. Since then, he has been continuously researched and developed new algorithms that are helpful for real applications in various fields. His research interests include data mining, privacy-preserving, big data analytics, parallel computing, social network, network infrastructure, and network security. In addition, he has over ten years of experience in design, implements network infrastructure systems, and security network systems.