http://dx.doi.org/10.62762/TISC.2026.252131

RESEARCH ARTICLE

ECC-MAKA-RFID: ECC-Based Mutual Authenticated Key Agreement for RFID Networks

Upendra Verma[1,*] **and Surendra Kumar Shukla**[2]

[1] Department of Computer Science, Mukesh Patel School of Technology Management and Engineering, SVKM's Narsee Monjee Institute of Management Studies (NMIMS), Shirpur 425405, India

[2] School of Computer Science Engineering and Technology, Bennett University, Greater Noida 201310, India

Abstract

Radio-frequency identification (RFID) is widely used in smart transportation, smart cities and healthcare, where tags and readers exchange identification data over an open wireless channel. Because this channel is public, weak authentication can leak sensitive information and leave RFID networks open to security and privacy attacks. This paper presents *ECC-MAKA-RFID*, a mutual authenticated key agreement scheme for RFID networks that uses elliptic curve cryptography (ECC) and a one-way hash function to authenticate a tag and a reader and to establish a session key over an insecure channel. The informal analysis shows that *ECC-MAKA-RFID* provides anonymity, untraceability and session-key agreement and resists replay, denial-of-service (DoS), man-in-the-middle (MITM) and impersonation attacks. The scheme is formally analyzed in the Real-or-Random (RoR) model and validated with the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool under the Dolev-Yao (DY) adversary model. Compared with four recent schemes, *ECC-MAKA-RFID* has the lowest computational cost, needs only two messages per session, and has a communication cost comparable to that of the existing schemes, which makes it suitable for resource-constrained RFID networks.

Keywords: Elliptic curve cryptography, adversary model, mutual authenticated key agreement, radio-frequency identification, RFID security.

1 Introduction

The Internet of Things (IoT) connects billions of devices, many of which have very limited computing and energy resources, so lightweight authentication is a recurring design requirement [1]. Radio-frequency identification (RFID) is one of the main enabling technologies of the IoT [2]. An RFID system consists of tags, readers and a back-end database server. Tags are attached to objects for automatic identification and tracking, and readers act as interrogators that identify the tags. A tag and a reader communicate over a wireless link to identify, authenticate and exchange information about the tagged object. RFID is used in smart grids, smart healthcare, smart cities and smart agriculture [2], where it helps improve productivity and protect valuable assets.

Submitted: 15 August 2026
Revised: 13 October 2026
Accepted: 25 October 2026
Published: 26 November 2026

Vol. 2, **No.** 4, 2026.
10.62762/TISC.2026.252131

***Corresponding author:**
✉ Upendra Verma
upendra4567@gmail.com

Citation
Verma, U., & Shukla, S. K. (2026). ECC-MAKA-RFID: ECC-Based Mutual Authenticated Key Agreement for RFID Networks. *ICCK Transactions on Information Security and Cryptography*, 2(4), 216–226.

© 2026 ICCK (Institute of Central Computation and Knowledge)

Because RFID tags are resource-constrained devices, they cannot run heavy cryptographic algorithms, and the security of RFID networks remains an active research topic [3]. An adversary can eavesdrop on, modify or replay tag-reader messages to obtain sensitive information. Secure authentication between tags and readers is therefore essential. Many schemes are built on symmetric-key cryptography, but they still have to solve the problem of establishing a shared secret key between tags and readers.

ECC-based authentication schemes have recently been proposed to address this problem in RFID applications, yet several of them remain vulnerable to common attacks. We therefore propose a new ECC-based authenticated key agreement scheme that provides anonymity and untraceability and that resists DoS, MITM, replay and impersonation attacks.

1.1 Problem Statement and Motivation

Mutual authenticated key agreement (MAKA) is required in RFID networks to secure communication and prevent unauthorized access. MAKA schemes are either symmetric-key based or asymmetric-key based. Several symmetric-key MAKA schemes have been developed [4–6]. Their main advantage is a lower computational overhead and therefore lower resource consumption. Their main drawback is key distribution: the two parties must share a secret key in advance. Asymmetric-key MAKA avoids this problem by using a public/private key pair. RSA and ECC are the most widely used asymmetric primitives, and ECC reaches a security level comparable to RSA with much smaller keys and lower computational cost, and the performance of both in hybrid encryption for secure data transmission has been compared in [7]. Since RFID tags have limited computing power, memory and storage, ECC is a suitable choice for RFID networks. Several ECC-based MAKA schemes have been proposed for RFID networks [8–11]. However, these schemes do not provide all of the required security properties, such as anonymity and untraceability, or they cannot resist common attacks such as MITM, replay, DoS and impersonation attacks. The main contributions of this paper are as follows:

- We propose *ECC-MAKA-RFID*, a lightweight authentication and key agreement scheme for resource-constrained RFID networks that keeps both computational and communication overheads low.
- We analyze the security of *ECC-MAKA-RFID* formally in the RoR model and validate it with the AVISPA tool.
- We compare *ECC-MAKA-RFID* with existing schemes on nine security features and on computational and communication cost.

1.2 System Model

The system model has three entities: tags, readers and a back-end server. Tags and readers exchange information over a wireless channel. In *ECC-MAKA-RFID*, the reader and the back-end server are treated as a single entity. A tag is activated by the reader's radio-frequency signal, and the reader relies on the tag to provide identification and authentication data. *ECC-MAKA-RFID* therefore focuses on securing the tag-reader channel. As in other timestamp-based RFID schemes, we assume that the tag and the reader keep loosely synchronized clocks so that the bound ΔT can be checked. Figure 1 shows the system model.

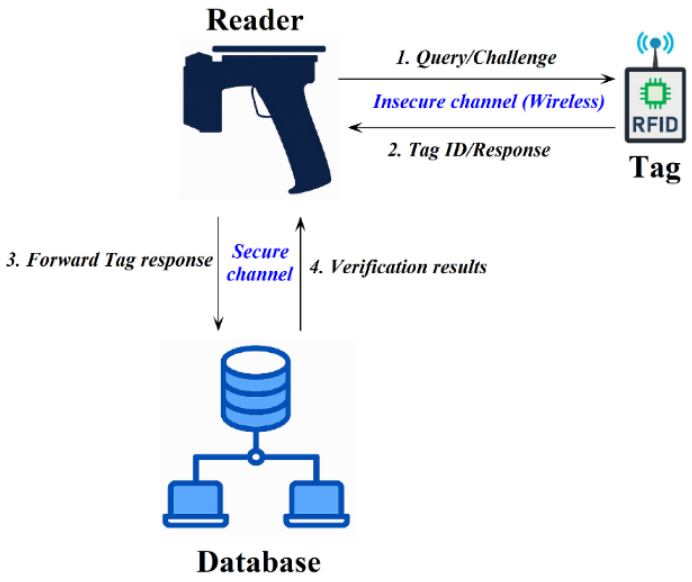


Figure 1. RFID system model.

1.3 Adversary Model

The security of *ECC-MAKA-RFID* is analyzed under the Dolev-Yao (DY) adversary model. Under this model, the adversary $\mathcal{A}$ fully controls the public channel and can

- intercept all messages exchanged between the tag and the reader;
- inject forged messages into the channel;
- modify, replay or drop any message transmitted between the tag and the reader.

Table 1. Summary of related work. PFS: perfect forward secrecy.

Scheme	Primitives	Weaknesses
Zhang and Qi [12], 2014	ECC	Does not provide PFS
Izza et al. [13], 2021	ECC	Cannot resist eavesdropping and brute-force attacks
Arslan et al. [14], 2021	ECC	Cannot resist replay attacks
Agrahari and Varma [15], 2021	ECC, hash	Does not provide PFS
Yao et al. [16], 2022	ECC	Cannot resist replay, impersonation and MITM attacks
Safkhani et al. [17], 2022	Bitwise operations	Cannot resist replay and impersonation attacks
Pirmoradian et al. [18], 2023	ECC	Cannot resist replay attacks
Wu et al. [19], 2024	ECC	Vulnerable to DoS attacks
Alghamdi [20], 2025	ECC	Vulnerable to replay and MITM attacks
Verma and Dhanare [21], 2025	ECC	Does not provide untraceability
Thota et al. [22], 2026	ECC	No formal security analysis
Zhu et al. [23], 2026	ECC	Vulnerable to desynchronization attacks

In the RoR analysis of Section 4, $\mathcal{A}$ may also issue *Corrupt* queries to obtain the long-term secrets stored in a tag or a reader.

1.4 Organization of the Paper

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents *ECC-MAKA-RFID*. Sections 4 and 5 analyze the scheme in the RoR model and with AVISPA, respectively. Section 6 gives the informal security analysis. Section 7 evaluates the performance of *ECC-MAKA-RFID*, and Section 8 concludes the paper.

2 Related Work

Many MAKA schemes have been proposed for RFID networks, based on different cryptographic primitives. Since *ECC-MAKA-RFID* is ECC based, this section focuses on ECC-based authentication schemes for RFID and for closely related resource-constrained IoT settings (e.g., body area networks, e-healthcare and smart grids), together with one ultra-lightweight RFID scheme that uses only bitwise operations [17]. For example, Farash et al. [24] proposed a provably secure ECC-based RFID authentication protocol for healthcare environments and showed that the earlier scheme of Zhang and Qi [12] does not provide forward secrecy. Table 1 summarizes these schemes and their weaknesses.

Table 1 shows that the existing schemes either miss some required security features or are vulnerable to common attacks. To address these weaknesses, we propose a new MAKA scheme for RFID networks.

3 Proposed Scheme

This section describes *ECC-MAKA-RFID*, which consists of three phases: system setup, registration, and authentication and key agreement. The notation is listed in Table 2.

Table 2. Notation used in *ECC-MAKA-RFID*.

Symbol	Meaning
T_i	ith tag
R_j	jth reader
RA	Registration authority
ID_T, ID_R	Identities of T_i and R_j
$E_q(a,b)$	Non-singular elliptic curve over $\mathbb{F}_q$; p is the order of the base point
GP	Generator (base) point of $E_q(a,b)$
PR_T, PU_T	Private and public keys of T_i
PR_R, PU_R	Private and public keys of R_j
$h(\cdot)$	One-way hash function with an l-bit output
T_T, T_R	Timestamps
ΔT	Maximum transmission delay
$\oplus$	XOR operation
$\|$	Concatenation
$\mathcal{A}$	Adversary

3.1 System Setup Phase

RA performs the following steps:

1. RA selects a non-singular elliptic curve $E_q(a,b)$ and a generator point $GP \in E_q(a,b)$ of prime order p.
2. RA selects a hash function $h(\cdot)$ and distributes $\{E_q(a,b), GP, h(\cdot)\}$ to T_i and R_j.

3.2 Registration Phase of T_i

1. T_i selects a random nonce n_T and its identity ID_T, computes $N_T = n_T \cdot GP$, and sends the registration request $\{ID_T, N_T\}$ to RA over a secure channel.

2. On receiving the request, RA checks whether ID_T is valid. If not, RA rejects the request and asks T_i to submit a new identity. Otherwise, RA generates a random nonce n_T^{RA}, computes the public key $PU_T = N_T + n_T^{RA} \cdot GP$, stores $\{ID_T, PU_T\}$ in its database, and sends $\{PU_T, n_T^{RA}\}$ to T_i over a secure channel.

3. T_i computes its private key $PR_T = n_T + n_T^{RA}$ and checks whether $PU_T = PR_T \cdot GP$. If not, T_i aborts the registration. Otherwise, once R_j has also been registered, RA sends $\{ID_R, PU_R\}$ to T_i over a secure channel, and T_i computes $X_R = PR_T \cdot PU_R$ and stores $\{ID_T, PR_T, X_R\}$. Figure 2 shows the registration of T_i.

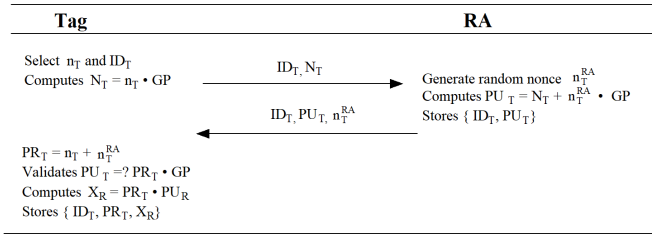


Figure 2. Registration of tag T_i.

3.3 Registration Phase of R_j

1. R_j selects a random nonce n_R and its identity ID_R, computes $N_R = n_R \cdot GP$, and sends $\{ID_R, N_R\}$ to RA over a secure channel.

2. RA checks whether ID_R is valid. If not, RA rejects the request and asks R_j to submit a new identity. Otherwise, RA generates a random nonce n_R^{RA}, computes $PU_R = N_R + n_R^{RA} \cdot GP$, stores $\{ID_R, PU_R\}$ in its database, and sends $\{PU_R, n_R^{RA}\}$ to R_j over a secure channel.

3. R_j computes its private key $PR_R = n_R + n_R^{RA}$ and checks whether $PU_R = PR_R \cdot GP$. If not, R_j aborts the registration. Otherwise, once T_i has also been registered, RA sends $\{ID_T, PU_T\}$ to R_j over a secure channel, and R_j computes $X_T = PR_R \cdot PU_T$ and stores $\{ID_R, PR_R, X_T\}$. Figure 3 shows the registration of R_j.

3.4 Authentication and Key Agreement Phase

Note that $X_R = PR_T PR_R \cdot GP = X_T$, so T_i and R_j share a static secret point after registration. For simplicity, the description considers one tag-reader

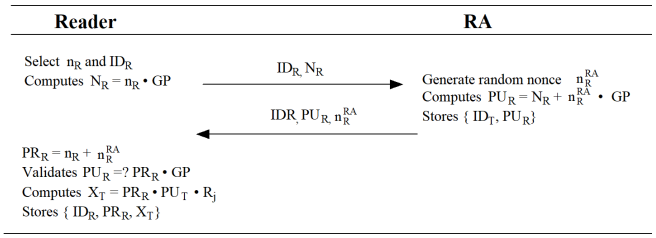


Figure 3. Registration of reader R_j.

pair. With several tags, R_j stores one secret point per registered tag and searches them to identify the tag, which adds a search cost that is not included in the evaluation of Section 7. The phase proceeds as follows (Figure 4).

1. T_i selects a random value v_T and a timestamp T_T, and computes $U_T = v_T \cdot PU_T$, $J_T = h(U_T \| X_R) \cdot GP$, $EID_T = ID_T \oplus h(J_T \| T_T)$ and $S_T = h(ID_T \| U_T \| J_T \| T_T)$. T_i sends the authentication request $\{EID_T, U_T, S_T, T_T\}$ to R_j.

2. On receiving the request at time T_T^*, R_j checks whether $|T_T^* - T_T| < \Delta T$. If so, R_j computes $J_R = h(U_T \| X_R) \cdot GP$, $ID_T^* = EID_T \oplus h(J_R \| T_T)$ and $S_T^* = h(ID_T^* \| U_T \| J_R \| T_T)$, and checks whether $S_T^* = S_T$. If not, the session is terminated.

3. R_j selects a random value v_R and a timestamp T_R, and computes $U_R = v_R \cdot PU_R$, the session key $SK_R = h(ID_T^* \| ID_R \| J_R \| U_T \| U_R)$ and $S_R = h(ID_R \| U_R \| J_R \| T_R)$.

4. R_j sends the authentication response $\{U_R, S_R, T_R\}$ to T_i.

5. On receiving the response at time T_R^*, T_i checks whether $|T_R^* - T_R| < \Delta T$. If so, T_i computes $S_R^* = h(ID_R \| U_R \| J_T \| T_R)$ and checks whether $S_R^* = S_R$. If not, the session is terminated.

6. Finally, T_i computes the session key $SK_T = h(ID_T \| ID_R \| J_T \| U_T \| U_R)$.

4 Formal Security Analysis in the RoR Model

The Real-or-Random (RoR) model is widely used to analyze authenticated key agreement protocols; for example, it was applied to a lightweight authentication mechanism for cloud-based IoT in [25]. In this model, $\mathcal{A}$ interacts with protocol participants through *Execute*, *Send*, *Corrupt* and *Test* queries. The goal of $\mathcal{A}$ is to decide whether the key returned by a *Test* query is the real session key or a random value. A protocol is secure if the advantage of $\mathcal{A}$ over random guessing is negligible.

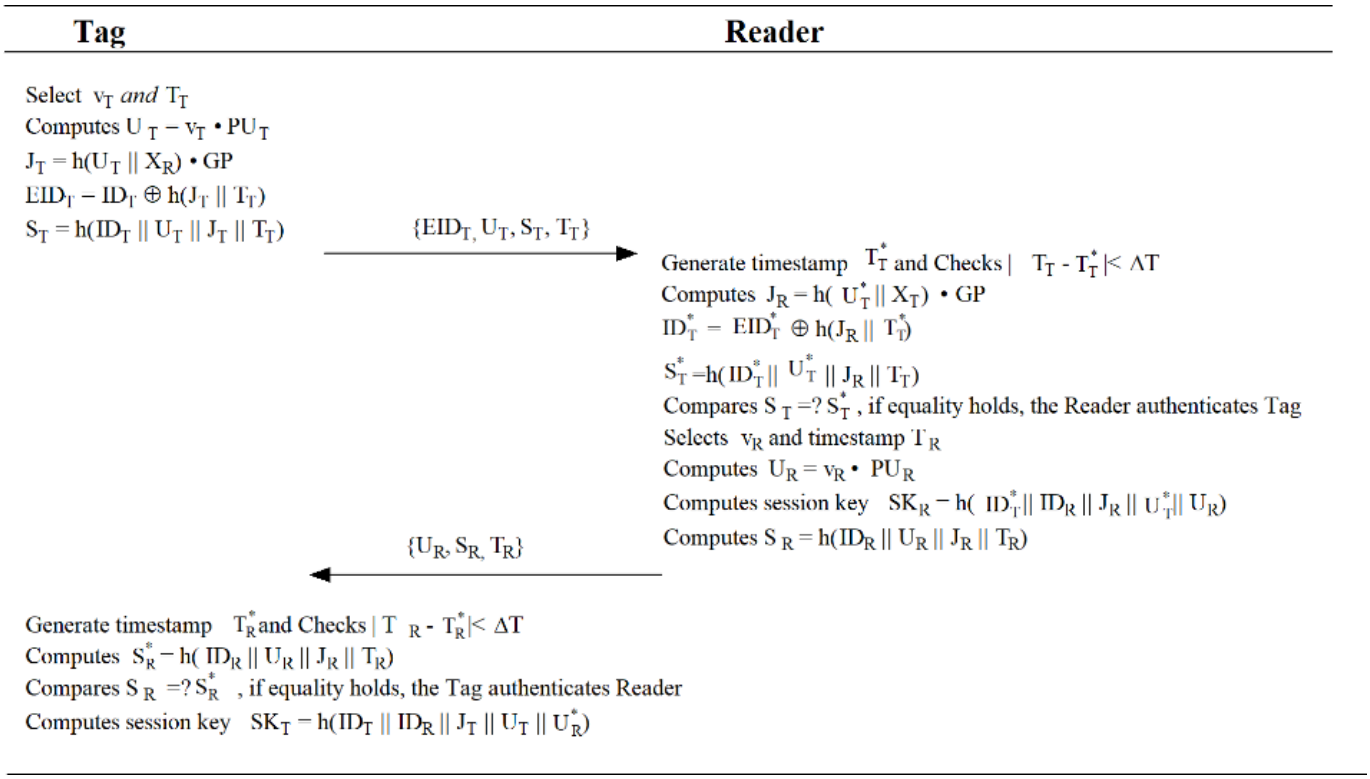


Figure 4. Authentication and key agreement phase of *ECC-MAKA-RFID*.

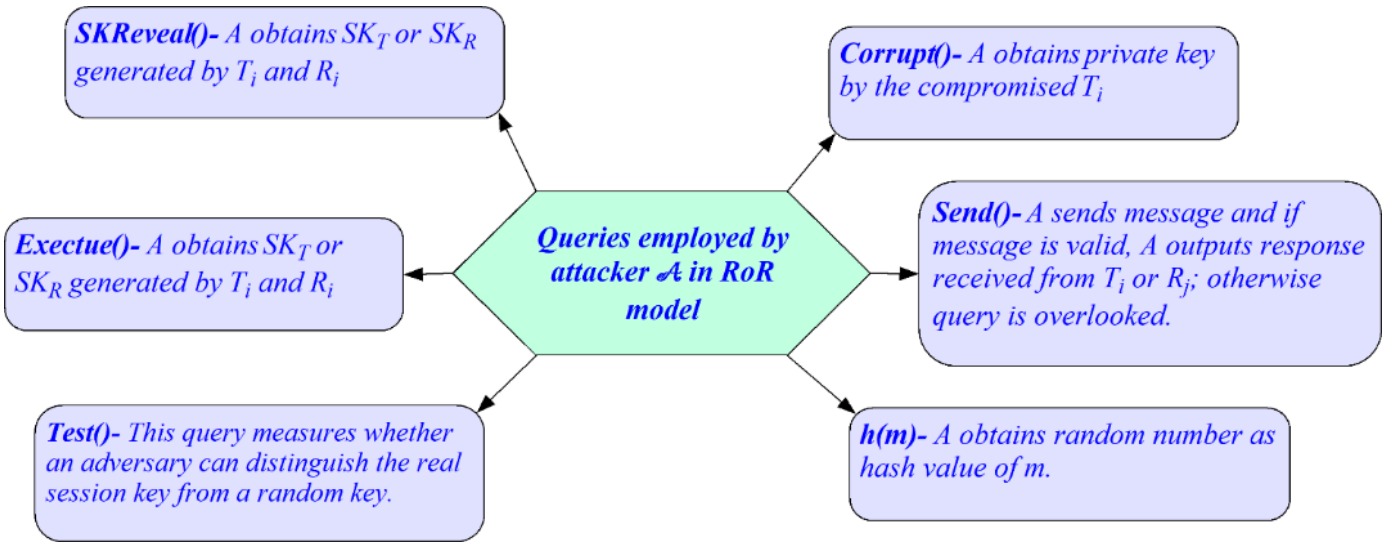


Figure 5. Oracle queries available to the adversary in the RoR model.

Participants. *ECC-MAKA-RFID* has two kinds of participants, the tag T_i and the reader R_j. Each participant may run several instances (oracles); an instance is denoted $Inst_i$.

States. Each instance can be in one of three states:

(a) *Empty*: the instance has not yet answered any input.

(b) *Accepted*: $Inst_i$ has received the last expected protocol message.

(c) *Fresh*: the session key between T_i and R_j has not been revealed to $\mathcal{A}$ by any query.

Partnering. T_i and R_j are partners if (a) they are in the accepted state, and (b) they have authenticated each other and share the same session key.

Adversary. $\mathcal{A}$ fully controls the public channel and can eavesdrop on, replay, intercept, delete or modify messages. Figure 5 lists the queries available to $\mathcal{A}$.

Semantic security. $\mathcal{A}$ tries to break the semantic security of *ECC-MAKA-RFID* by using the queries

$Send(Inst_i, Msg)$, $SKReveal(Inst_i)$, $Corrupt(Inst_i)$, $Execute(Inst_i)$ and $Test(Inst_i)$, and then outputs a guess c' of the hidden bit c used in the *Test* query. Let $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{P}}(t)$ denote the advantage of $\mathcal{A}$ in breaking the semantic security of *ECC-MAKA-RFID* (denoted $\mathcal{P}$) within polynomial time t, and let Succ_i be the event that $\mathcal{A}$ guesses c correctly in game G_i. Then

$$\mathrm{Adv}_{\mathcal{A}}^{\mathcal{P}}(t) = |2\mathrm{Pr}[c' = c] - 1| = |2\mathrm{Pr}[\mathrm{Succ}] - 1|. \quad (1)$$

Lemma 1. Let Q_s, Q_h and Q_e be the numbers of *Send*, hash and *Execute* queries made by $\mathcal{A}$ within time t, and let l be the output length of $h(\cdot)$. Then

$$\mathrm{Adv}_{\mathcal{A}}^{\mathcal{P}}(t) \le \frac{Q_h^2}{2^l} + \frac{(Q_s+Q_e)^2}{p-1} + \frac{(Q_s+2Q_h)^2 + (Q_s+Q_h)^2}{2^{l-1}} + 2\,\mathrm{Adv}_{\mathcal{A}}^{\mathrm{ECDLP}}(t). \quad (2)$$

Proof. The proof uses a sequence of games G_i ($0 \le i \le 4$), each of which simulates the execution of *ECC-MAKA-RFID* and answers the queries of $\mathcal{A}$.

G_0: This game is the real attack in the RoR model. Hence

$$\mathrm{Adv}_{\mathcal{A}}^{\mathcal{P}}(t) = |2\mathrm{Pr}[\mathrm{Succ}_0] - 1|. \quad (3)$$

G_1: $\mathcal{A}$ issues $Execute(Inst_i)$ queries to eavesdrop on the messages $\{EID_T, U_T, S_T, T_T\}$ and $\{U_R, S_R, T_R\}$, and then a $Test(Inst_i)$ query. Since SK_T and SK_R depend on $J_T = J_R$, which is derived from the secret point $X_R = X_T$, $\mathcal{A}$ cannot compute the session key from the intercepted messages. Therefore,

$$|\mathrm{Pr}[\mathrm{Succ}_1] - \mathrm{Pr}[\mathrm{Succ}_0]| = 0. \quad (4)$$

G_2: $\mathcal{A}$ tries to forge valid messages by issuing *Send* and hash queries. The simulation fails only if a collision occurs. By the birthday paradox, (a) a collision among the random values occurs with probability at most $(Q_s + Q_e)^2/[2(p-1)]$, and (b) a collision among hash outputs occurs with probability at most $Q_h^2/2^{l+1}$. Games G_1 and G_2 are indistinguishable unless a collision occurs, so

$$|\mathrm{Pr}[\mathrm{Succ}_2] - \mathrm{Pr}[\mathrm{Succ}_1]| \le \frac{Q_h^2}{2^{l+1}} + \frac{(Q_s+Q_e)^2}{2(p-1)}. \quad (5)$$

G_3: $\mathcal{A}$ tries to guess the secret values shared by T_i and R_j in order to forge a valid message. Having obtained the request $\{EID_T, U_T, S_T, T_T\}$, $\mathcal{A}$ can produce a valid request through hash queries with probability at most $(Q_s+2Q_h)^2/2^l$; similarly, a valid response $\{U_R, S_R, T_R\}$ can be produced with probability at most $(Q_s+Q_h)^2/2^l$. Hence

$$|\mathrm{Pr}[\mathrm{Succ}_3] - \mathrm{Pr}[\mathrm{Succ}_2]| \le \frac{(Q_s+2Q_h)^2 + (Q_s+Q_h)^2}{2^l}. \quad (6)$$

G_4: $\mathcal{A}$ issues $Corrupt(Inst_i)$ queries to one party and uses the intercepted messages to try to derive the session key. Recovering the ephemeral scalar v_T from U_T or v_R from U_R requires solving the elliptic curve discrete logarithm problem (ECDLP). Therefore,

$$|\mathrm{Pr}[\mathrm{Succ}_4] - \mathrm{Pr}[\mathrm{Succ}_3]| \le \mathrm{Adv}_{\mathcal{A}}^{\mathrm{ECDLP}}(t). \quad (7)$$

After all queries, $\mathcal{A}$ has no information about the hidden bit c, so

$$\mathrm{Pr}[\mathrm{Succ}_4] = \frac{1}{2}. \quad (8)$$

From Eq. (3) and Eq. (8), $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{P}}(t) = 2|\mathrm{Pr}[\mathrm{Succ}_0] - \mathrm{Pr}[\mathrm{Succ}_4]|$. Applying the triangle inequality to Eqs. (4)–(7) gives Eq. (2), which proves Lemma 1.

5 Formal Security Validation with AVISPA

We use AVISPA to validate *ECC-MAKA-RFID* formally. AVISPA is a widely used tool for verifying the security of cryptographic, key agreement and authentication protocols [26]. It checks whether a protocol meets security goals such as secrecy and authentication under the Dolev-Yao intruder, who fully controls the public channel. Protocols are specified in the High-Level Protocol Specification Language (HLPSL), and the tool can detect attacks such as replay, impersonation, MITM and message modification before deployment [27]. The tool is publicly available at http://www.avispa-project.org/. Figure 6 shows its web interface.

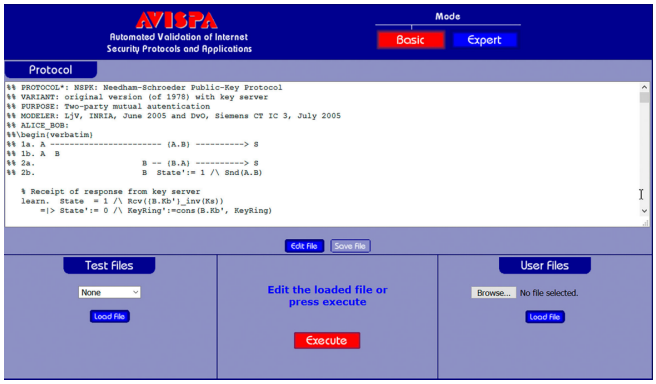


Figure 6. Web interface of the AVISPA tool.

Simulation setup. The simulation was run on an Intel Core i5-8265U CPU @ 1.60 GHz with 8 GB of RAM, an x64-based processor and 64-bit Windows 10; the Security Protocol ANimator (SPAN) and AVISPA environment ran in an Ubuntu virtual machine (Figure 7).

The security goals of *ECC-MAKA-RFID* are specified in HLPSL and translated into the intermediate format (IF), which the AVISPA back-ends analyze. We use the On-the-Fly Model Checker (OFMC) and the Constraint-Logic-based Attack Searcher (CL-AtSe) back-ends. The SAT-based Model Checker (SATMC) and the Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP) back-ends are not used because they do not support the algebraic properties of XOR.

Simulation results. As shown in Figure 7, OFMC visits 84 nodes at a search depth of 6 plies, and CL-AtSe analyzes 6 states with a translation time of 0.28 s. Both back-ends report the protocol as SAFE, which indicates that *ECC-MAKA-RFID* resists the attacks modelled by the DY intruder.

6 Informal Security Analysis

F1: Anonymity. The identities of T_i and R_j are never sent in plaintext during authentication. ID_T is masked as $EID_T = ID_T \oplus h(J_T \,\|\, T_T)$, and $\mathcal{A}$ cannot recover ID_T from EID_T without J_T. Hence *ECC-MAKA-RFID* provides anonymity.

F2: Mutual authentication. R_j checks $|T_T^* - T_T| < \Delta T$ and verifies $S_T^* = S_T$; T_i checks $|T_R^* - T_R| < \Delta T$ and verifies $S_R^* = S_R$. Otherwise the session is terminated. Hence *ECC-MAKA-RFID* provides mutual authentication.

F3: Resistance to impersonation attacks. To impersonate T_i, $\mathcal{A}$ must produce a request $\{EID_T, U_T, S_T, T_T\}$ that R_j accepts. This requires J_T, which depends on the secret point X_R derived from PR_T. Without it, $\mathcal{A}$ cannot produce a valid request in polynomial time. Hence *ECC-MAKA-RFID* resists impersonation attacks.

F4: Key agreement. T_i computes $SK_T = h(ID_T \,\|\, ID_R \,\|\, J_T \,\|\, U_T \,\|\, U_R)$ and R_j computes $SK_R = h(ID_T^* \,\|\, ID_R \,\|\, J_R \,\|\, U_T \,\|\, U_R)$. Since $X_R = X_T$, we have $J_T = J_R$ and therefore $SK_T = SK_R$.

F5: Untraceability. The values U_T, EID_T, S_T, U_R and S_R change in every session because they depend on fresh random values and timestamps. $\mathcal{A}$ therefore cannot link two sessions to the same tag. Hence *ECC-MAKA-RFID* provides untraceability.

F6: Resistance to replay attacks. Each message carries a timestamp (T_T or T_R) that is covered by S_T or S_R. A message replayed after ΔT fails the freshness check $|T^* - T| < \Delta T$, and changing the timestamp invalidates S_T or S_R. A message replayed within ΔT passes the check, but $\mathcal{A}$ cannot derive the session key from the reader's response because v_T and J_T remain unknown; R_j can additionally cache recent values of U_T and reject duplicates. Hence *ECC-MAKA-RFID* resists replay attacks.

F7: Resistance to key compromise attacks. Assume that $\mathcal{A}$ learns ID_T and PU_T of T_i. $\mathcal{A}$ still cannot impersonate R_j to T_i or compute SK_T, because v_T is fresh in every session and PR_T remains secret.

F8: Resistance to MITM attacks. Suppose $\mathcal{A}$ captures the messages between T_i and R_j and tries to modify them so that either party accepts a forged message. To modify $\{EID_T, U_T, S_T, T_T\}$ consistently, $\mathcal{A}$ needs J_T, which depends on the secret X_R. Hence *ECC-MAKA-RFID* resists MITM attacks.

F9: Resistance to DoS attacks. *ECC-MAKA-RFID* uses the timestamps T_T and T_R so that messages outside the window ΔT are rejected before any point multiplication is performed. A message inside the window still costs the reader one point multiplication (for J_R) before S_T can be checked, so *ECC-MAKA-RFID* limits, but does not eliminate, the cost an adversary can impose by flooding a reader with requests.

7 Performance Evaluation

7.1 Communication Cost

Table 3. Communication cost of *ECC-MAKA-RFID* and related schemes.

Scheme	Cost $T_i \leftrightarrow R_j$ (bits)	Messages
[15]	1280	3
[28]	1184	3
[29]	1184	3
[30]	1120	3
ECC-MAKA-RFID	1184	2

We assume 320-bit ECC points, 160-bit identities and random numbers, 32-bit timestamps, 128-bit symmetric encryption/decryption outputs and 160-bit hash outputs. These sizes correspond to a 160-bit curve and a 160-bit hash, which give roughly 80-bit security; larger parameters (e.g., a 256-bit curve and SHA-256)

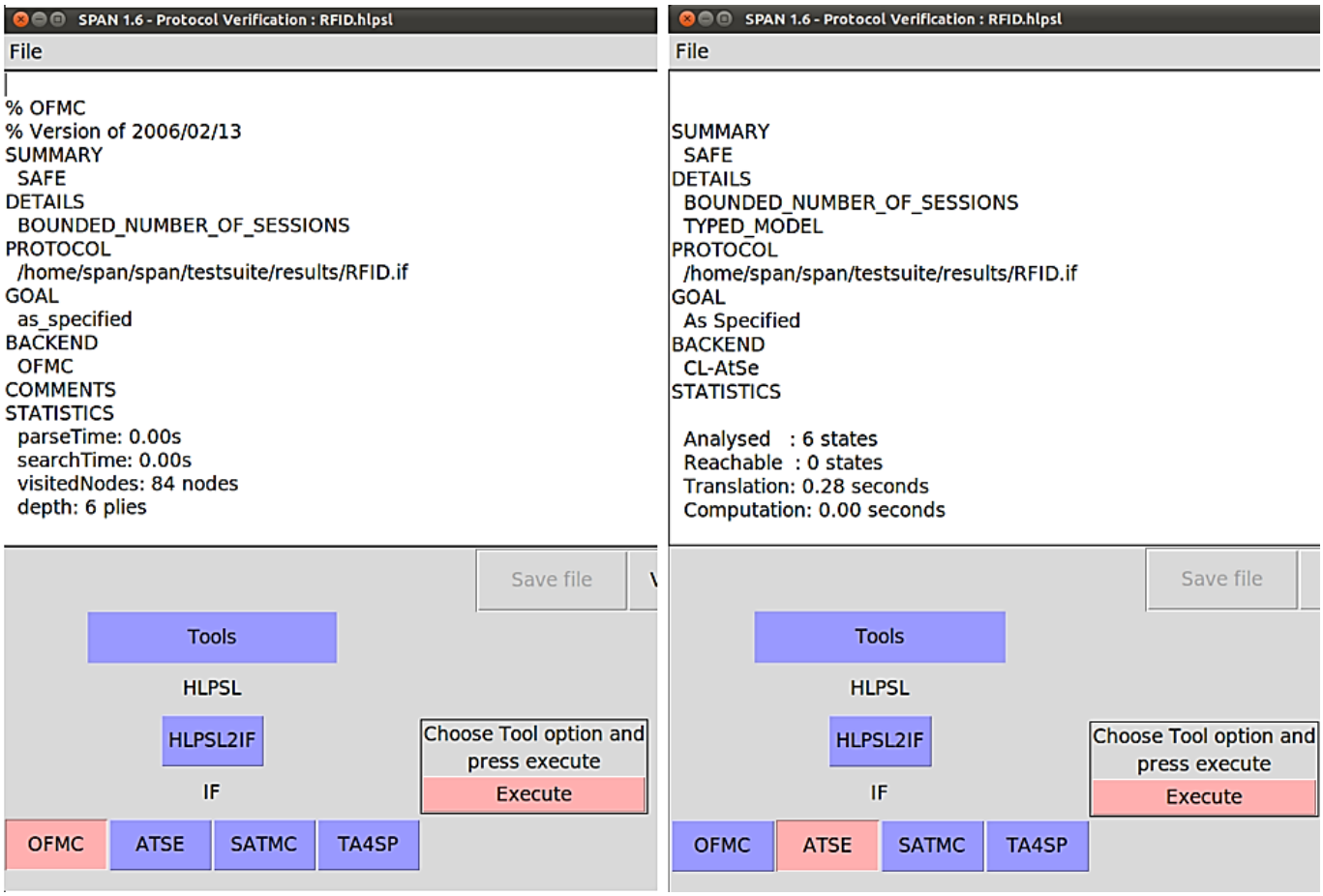


Figure 7. Simulation results of the OFMC and CL-AtSe back-ends.

Table 4. Computational cost of *ECC-MAKA-RFID* and related schemes.

Scheme	Operations of T_i	T_i (ms)	Operations of R_j	R_j (ms)	Total operations	Total (ms)
[15]	$4T_{em} + T_h + T_{e/d}$	1.669	$4T_{em} + T_h + T_{e/d}$	1.669	$8T_{em} + 2T_h + 2T_{e/d}$	3.338
[17]	$3T_{em} + 6T_{ea} + 3T_h$	6.177	$4T_{em} + 8T_{ea} + 3T_h$	8.087	$7T_{em} + 14T_{ea} + 6T_h$	14.264
[29]	$3T_{em} + 4T_h + T_{e/d}$	1.766	$2T_{em} + 5T_h + T_{e/d}$	1.565	$5T_{em} + 9T_h + 2T_{e/d}$	3.331
[30]	$3T_{em} + 3T_{ea} + 2T_{e/d}$	3.630	$4T_{em} + 4T_{ea} + 2T_{e/d}$	4.760	$7T_{em} + 7T_{ea} + 4T_{e/d}$	8.390
ECC-MAKA-RFID	$2T_{em} + 5T_h$	1.445	$2T_{em} + 5T_h$	1.445	$4T_{em} + 10T_h$	2.890

would raise the communication cost of all compared schemes. The messages exchanged between T_i and R_j cost:

- Request $\{EID_T, U_T, S_T, T_T\}$ from T_i to R_j: $160 + 320 + 160 + 32 = 672$ bits;
- Response $\{U_R, S_R, T_R\}$ from R_j to T_i: $320 + 160 + 32 = 512$ bits.

ECC-MAKA-RFID thus exchanges two messages with a total cost of $672 + 512 = 1184$ bits. Table 3 and Figure 8 compare this cost with existing schemes.

As shown in Table 3 and Figure 8, the communication

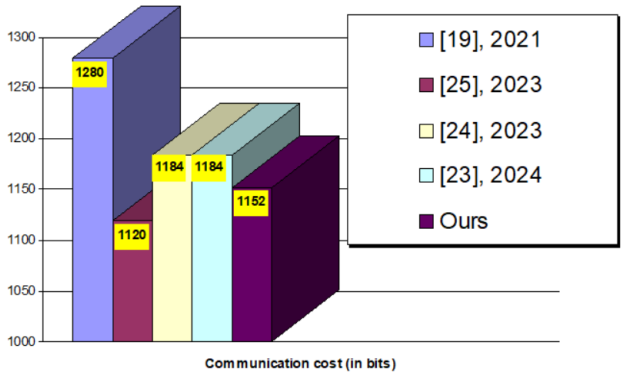


Figure 8. Comparison of communication cost.

cost of *ECC-MAKA-RFID* is lower than that of [15], equal to that of [28, 29] and slightly higher than that of [30]. However, *ECC-MAKA-RFID* completes authentication with two messages instead of three, which saves one message on the tag-reader link.

7.2 Computational Cost

Let T_h, T_{em}, T_{ea} and $T_{e/d}$ denote the time of a hash operation, an ECC point multiplication, an ECC point addition and a symmetric encryption/decryption, respectively. Following [31], we use $T_{em} \approx 0.35$ ms, $T_{ea} \approx 0.78$ ms and $T_h \approx 0.149$ ms, and we take $T_{e/d} \approx 0.12$ ms. XOR and concatenation are ignored because their cost is negligible. In *ECC-MAKA-RFID*, T_i performs two point multiplications and five hash operations, i.e., $2T_{em} + 5T_h \approx 1.445$ ms; R_j performs the same operations, also ≈ 1.445 ms. The total cost is $4T_{em} + 10T_h \approx 2.89$ ms. Table 4 and Figure 9 compare *ECC-MAKA-RFID* with existing schemes.



Figure 9. Comparison of computational cost.

As shown in Figure 9, the computational cost of the schemes in [15, 17, 29, 30] is higher than that of *ECC-MAKA-RFID*. *ECC-MAKA-RFID* therefore has the lowest computational cost among the compared schemes.

7.3 Comparison of Security Features

Table 5 compares the security features of *ECC-MAKA-RFID* with those of related schemes [13, 15, 17, 20, 30]. *ECC-MAKA-RFID* satisfies all nine features considered, whereas each of the compared schemes either fails to provide or does not consider at least one of them.

8 Conclusion

This paper presented *ECC-MAKA-RFID*, an ECC-based mutual authenticated key agreement scheme for RFID networks that lets a tag and a reader authenticate each other and agree on a session key. The scheme was analyzed informally against common attacks, analyzed formally in the RoR model, and validated with AVISPA. Compared with four recent schemes, *ECC-MAKA-RFID* has the lowest computational cost and needs fewer messages, with a comparable communication cost. In future work, we plan to implement *ECC-MAKA-RFID* on real RFID hardware to evaluate its practicality, scalability and performance, and to study post-quantum cryptographic techniques that can protect it against quantum-capable adversaries.

Table 5. Comparison of security features.

Scheme	F1	F2	F3	F4	F5	F6	F7	F8	F9
[13]	✓	✓	•	✓	•	✓	✓	•	•
[15]	✗	✓	•	✓	✗	✓	✗	✓	•
[17]	✗	✓	✗	✓	•	✗	✓	•	•
[30]	✓	✓	✗	✓	✓	✓	✓	✗	•
[20]	✓	✓	✓	✓	✓	✗	•	✗	•
ECC-MAKA-RFID	✓	✓	✓	✓	✓	✓	✓	✓	✓

F1: anonymity; F2: mutual authentication; F3: resistance to impersonation; F4: key agreement; F5: untraceability; F6: resistance to replay; F7: resistance to key compromise; F8: resistance to MITM; F9: resistance to DoS. ✓: provided/resisted; ✗: not provided/not resisted; •: not considered in the original paper.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

AI Use Statement

The authors declare that no generative AI was used in the preparation of this manuscript.

Ethical Approval and Consent to Participate

Not applicable. This study did not involve human participants or animals.

References

[1] Khurana, A., Gopikrishnan, S., Konda, S. R., & Kokila, M. (2026). Secure and efficient authentication architecture for IoT devices in resource-limited

networks. *ICCK Transactions on Information Security and Cryptography, 2*(1), 16-28. [CrossRef]

[2] Eteng, A. A., Abdul Rahim, S. K., & Leow, C. Y. (2025). RFID in the Internet of Things. In *Internet of Things A to Z: Technologies and Applications* (2nd ed., pp. 67-88). Wiley-IEEE Press. [CrossRef]

[3] Kumar, V., & Das, S. K. (2025). Enhancing security in IIoT: RFID authentication protocol for edge computing and blockchain-enabled supply chain. *Cyber Security and Applications, 3*, 100087. [CrossRef]

[4] Mansoor, K., Ghani, A., Chaudhry, S. A., Shamshirband, S., Ghayyur, S. A. K., & Mosavi, A. (2019). Securing IoT-based RFID systems: A robust authentication protocol using symmetric cryptography. *Sensors, 19*(21), 4752. [CrossRef]

[5] Gao, L., Zhang, L., & Ma, M. (2017). Low cost RFID security protocol based on rabin symmetric encryption algorithm. *Wireless Personal Communications, 96*(1), 683-696. [CrossRef]

[6] Sultan, A., Hassan, M., Mansoor, K., & Ahmed, S. S. (2021). Securing IoT enabled RFID based object tracking systems: A symmetric cryptography based authentication protocol for efficient smart object tracking. In *2021 International Conference on Communication Technologies* (*ComTech*) (pp. 7-12). IEEE. [CrossRef]

[7] Saydahd, S. J., Muhammed, R. K., Hassan, S. A., & Aladdin, A. M. (2025). A comparative performance evaluation of hybrid encryption techniques using ECC, RSA, AES, and ChaCha20 for secure data transmission. *Iraqi Journal of Industrial Research, 12*(2), 157-172. [CrossRef]

[8] Timouhin, H., Amounas, F., & Badiy, M. (2025). Efficient ECC-based RFID authentication for enhanced IoT security. In *Recent advances in internet of things security* (pp. 39-46). CRC Press. [CrossRef]

[9] Alamr, A. A., Kausar, F., Kim, J., & Seo, C. (2018). A secure ECC-based RFID mutual authentication protocol for internet of things. *The Journal of Supercomputing, 74*(9), 4281-4294. [CrossRef]

[10] Verma, V. (2026). Advanced RFID authentication over elliptical curves: Ensuring security and efficiency. *Serbian Journal of Electrical Engineering, 23*(1), 57-72. [CrossRef]

[11] Naeem, M., Chaudhry, S. A., Mahmood, K., Karuppiah, M., & Kumari, S. (2020). A scalable and secure RFID mutual authentication protocol using ECC for Internet of Things. *International Journal of Communication Systems, 33*(13), e3906. [CrossRef]

[12] Zhang, Z., & Qi, Q. (2014). An efficient RFID authentication protocol to enhance patient medication safety using elliptic curve cryptography. *Journal of Medical Systems, 38*(5), 47. [CrossRef]

[13] Izza, S., Benssalah, M., & Drouiche, K. (2021). An enhanced scalable and secure RFID authentication protocol for WBAN within an IoT environment. *Journal of Information Security and Applications, 58*, 102705. [CrossRef]

[14] Arslan, A., Çolak, S. A., & Ertürk, S. (2021). A secure and privacy friendly ECC based RFID authentication protocol for practical applications. *Wireless Personal Communications, 120*(4), 2653-2691. [CrossRef]

[15] Agrahari, A. K., & Varma, S. (2021). A provably secure RFID authentication protocol based on ECQV for the medical internet of things. *Peer-to-Peer Networking and Applications, 14*(3), 1277-1289. [CrossRef]

[16] Yao, H., Yan, Q., Fu, X., Zhang, Z., & Lan, C. (2022). ECC-based lightweight authentication and access control scheme for IoT E-healthcare. *Soft Computing, 26*(9), 4441-4461. [CrossRef]

[17] Safkhani, M., Rostampour, S., Bendavid, Y., Sadeghi, S., & Bagheri, N. (2022). Improving RFID/IoT-based generalized ultra-lightweight mutual authentication protocols. *Journal of Information Security and Applications, 67*, 103194. [CrossRef]

[18] Pirmoradian, F., Safkhani, M., & Dakhilalian, S. M. (2023). ECCPWS: An ECC-based protocol for WBAN systems. *Computer Networks, 224*, 109598. [CrossRef]

[19] Wu, Y., Guo, H., Han, Y., Li, S., & Liu, J. (2024). A security-enhanced authentication and key agreement protocol in smart grid. *IEEE Transactions on Industrial Informatics, 20*(9), 11449-11457. [CrossRef]

[20] Alghamdi, A. M. (2025). Design and analysis of lightweight and robust authentication protocol for securing the resource constrained IIoT environment. *PLOS ONE, 20*(2), e0318064. [CrossRef]

[21] Verma, U., & Dhanare, R. (2025). A robust ECC-based authenticated key agreement protocol for wireless body area networks. *Concurrency and Computation: Practice and Experience, 37*(27-28), e70361. [CrossRef]

[22] Thota, A., Babu, C. J., & Rawat, G. S. (2026, August). A secure and lightweight ECC-based authentication framework for IoT environment. In *2026 6th International Conference on Soft Computing for Security Applications* (*ICSCSA*) (pp. 420-425). IEEE. [CrossRef]

[23] Zhu, R., Ji, Z. Y., & Liu, W. R. (2026). A secured RFID authentication protocol in telecare medicine information system. *International Journal of Network Security, 28*(2), 503-510. [CrossRef]

[24] Farash, M. S., Nawaz, O., Mahmood, K., Chaudhry, S. A., & Khan, M. K. (2016). A provably secure RFID authentication protocol based on elliptic curve for healthcare environments. *Journal of Medical Systems, 40*, 165. [CrossRef]

[25] Wazid, M., Das, A. K., Bhat, V., & Vasilakos, A. V. (2020). LAM-CIoT: Lightweight authentication mechanism in cloud-based IoT environment. *Journal of Network and Computer Applications, 150*, 102496. [CrossRef]

[26] Belfaik, Y., Lotfi, Y., Sadqi, Y., & Safi, S. (2024, May). A

comparative study of protocols' security verification tools: Avispa, Scyther, ProVerif, and Tamarin. In *International Conference on Digital Technologies and Applications* (pp. 118-128). Cham: Springer Nature Switzerland. [CrossRef]

[27] Vigano, L. (2006). Automated security protocol analysis with the AVISPA tool. *Electronic Notes in Theoretical Computer Science, 155*, 61-86. [CrossRef]

[28] Timouhin, H., Amounas, F., & Azrour, M. (2023). New ECC-based IoT authentication protocol for securing RFID systems. *SN Computer Science, 4*(6), 785. [CrossRef]

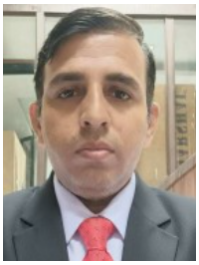
[29] Nikooghadam, M., Shahriari, H. R., & Saeidi, S. T. (2023). HAKECC: Highly efficient authentication and key agreement scheme based on ECDH for RFID in IOT environment. *Journal of Information Security and Applications, 76*, 103523. [CrossRef]

[30] Prakash, M., & Ramesh, K. (2024). ECAUT: ECC-infused efficient authentication for internet of things systems based on zero-knowledge proof. *The Journal of Supercomputing, 80*(17), 25640-25667. [CrossRef]

[31] Alzahrani, N. (2025). Security importance of edge-IoT ecosystem: An ECC-based authentication scheme. *PLOS ONE, 20*(6), e0322131. [CrossRef]



Upendra Verma is an Assistant Professor in the Department of Computer Science, Mukesh Patel School of Technology Management and Engineering, SVKM's NMIMS, Shirpur Campus, Maharashtra, India. He received the B.E. and M.E. degrees in computer science and engineering from RGTU Bhopal and the Ph.D. degree in computer science and engineering from GLA University, Mathura, India. He completed postdoctoral research in computer science and engineering at Lincoln University College, Petaling Jaya, Malaysia. His research interests include cyber security, fog computing and post-quantum cryptography. He reviews for many journals published by Springer, Elsevier, Wiley and IEEE. (Email: upendra.verma@nmims.edu)



Surendra Kumar Shukla is an Associate Professor in the School of Computer Science Engineering and Technology, Bennett University, Greater Noida, Uttar Pradesh, India. He received the Ph.D. degree in computer science and engineering from DAVV University, Indore, India. His research interests include blockchain, machine learning and information security. (Email: dr.surendrakshukla@gmail.com)