

Rough set method-cloud internet of things: a two-degree verification scheme for security in cloud-internet of things

Sheeba MaryJohn Rukmony¹, Suganthi Gnanamony²

¹St. Xavier's College (Autonomous), Affiliated to Manonmaniam Sundaranar University, Tirunelveli, India

²Department of Computer Science, Women's Christian College, Nagercoil, India

Article Info

Article history:

Received Feb 2, 2022

Revised Sep 15, 2022

Accepted Oct 13, 2022

Keywords:

Data sharing

Internet of things

Registered authority

Rough set machine

Security

ABSTRACT

The quick development of innovations and increasing use of the internet of things (IoT) in human life brings numerous challenges. It is because of the absence of adequate capacity resources and tremendous volumes of IoT information. This can be resolved by a cloud-based architecture. Consequently, a progression of challenging security and privacy concerns has emerged in the cloud based IoT context. In this paper, a novel approach to providing security in cloud based IoT environments is proposed. This approach mainly depends on the working of rough set rules for guaranteeing security during data sharing (rough set method-cloud IoT (RSM-CIoTD)). The proposed RSM-CIoTD conspire guarantees secure communication between the user and cloud service provider (CSP) in a cloud based IoT. To manage unauthorized users, an RSM-CIoTD scheme utilizes a registered authority which plays out a two-degree confirmation between the network substances. The security and privacy appraisal techniques utilize minimum and maximum trust benefits of past communication. The experiments show that our proposed system can productively and safely store the cloud service while outperforming other security methods.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Sheeba MaryJohn Rukmony

St. Xavier's College (Autonomous), Affiliated to Manonmaniam Sundaranar University

Abishekapatti, Tirunelveli-627012, India

Email: sheebasjustus@gmail.com.

1. INTRODUCTION

In quite a few years, smart city innovation will make networks more viable and useful, as metropolitan populaces are projected to grow worldwide [1]. Internet of things (IoT) is the quickly developing data innovation paradigm which is immensely conveyed in urban areas [2], and smart homes [3], [4]. With the fast development of IoT applications and gadgets, digital attacks will likewise be improved and represent a more serious threat to security and privacy than at any other time [5]. The ascent of cloud computing (CC) has given an elective arrangement, giving the IoT for all intents and purposes boundless wellspring of processing power, effectively available through the web, with better strength and at a lower cost [6], [7]. With new advancements in CC proceeding to push beyond the norm, nonetheless, different security dangers or stored data should be considered [8]–[12].

Motivated by an expanding number of weaknesses, attacks, and data leaks in IoT-based devices, researchers developed various methodologies for security and privacy-protecting in the cloud area [13], [14]. A cloud-based medical care framework was proposed in [15]. In [16], a framework planned to incorporate savvy IoT gadgets to permit admittance to patient data by means of the web utilizing cloud figuring. Four-venture engineering was intended for e-Wellbeing frameworks in [17]. An IoT-based body sensor network using lightweight remote sensor nodes with data sharing through cloud processing was proposed in [18]. The

benefits of cloud facilitating administrations were summed up in [19]. The presumption of a safe cloud is proper with regard to correspondence models talked about in [20]. To make a more trustworthy framework, Shabut *et al.* [21] distinguished the noxious hubs dependent on their behavior and further developed bundle conveyance through a multi-bounce hand-off network. A trust management model (TMM) was proposed to assess the trustworthiness of hubs through beta dissemination [22].

With the intention to ensure security and privacy in a cloud based IoT environment, we summarize the contributions of this research. We proposed a two-degree verification framework based on a rough set method for ensuring security and privacy in cloud-IoT (CIoTD). We provide a detailed security analysis to prove the importance of rough set method-cloud IoT (RSM-CIoTD). We also compared the proposed RSM-CIoTD scheme with some closely related existing schemes to show that the proposed RSM-CIoTD scheme offers an improved trade-off between the security and functionality features, in terms of communication and computation costs.

2. THE PROPOSED METHOD FOR SECURITY IN CLOUD-INTERNET OF THINGS

The proposed RSM-CIoTD scheme uses a two-degree verification scheme using authentication and trust, as key elements of security. The proposed RSM-CIoTD scheme, guarantees the veracity of transmitted messages using 1st degree verification, whereas 2nd degree verification guarantees the acceptability of users before originating communication with other entities. The RSM-CIoTD scheme depends on the minimum and maximum trust values of past communication. To this end, the users with a precise level of trust score or more are considered as legitimate and can communicate with other users, otherwise, it halts the data communication process.

2.1. Network model

The structure of the proposed RSM-CIoTD network model comprises three layers namely observation layer, IoT layer, and cloud layer. The observation layer is at the lower layer that grasps data from this present reality. It communicates straightforwardly with the climate where the use of IoT is executed. IoT sensors ($S(iot)_i$) are deployed in the IoT layer. Suppose a user (US_i) wants to access $S(iot)_i$ directly US_i and $S(iot)_i$ need mutual authentication among each other. After mutual authentication, both US_i and $S(iot)_i$ create a session key for future secure communication. Since all $S(iot)_i$ are associated with the cloud server (CS), they can detect and direct data to CS for capacity and further handling through gateway G_{way} nodes safely to the CSs utilizing some deterministic key administration plans. To get the communication between a US_i and $S(iot)_i$, they can commonly verify with one another and furthermore build up a mysterious session key. In cloud layer, there is a cloud service provider (CSP) and a registered authority (RA) which creates the essential authorizations for $S(iot)_i$ and G_{way} , and then records the authorizations in their memory prior to their deployment in the network.

2.2. Adversary model

In the proposed RSM-CIoTD, the communication network is open, and the conveying endpoint parties like US_i and $S(iot)_i$ are not dependable. An attacker (A_t) would then be able to listen in the traded messages. Indeed, even the messages might be erased or changed during the transmission. The physical catching of $S(iot)_i$ is conceivable by A_t . Thus, if truly catches some $S(iot)_i$, and the client US_i can separate qualifications from the memory of $S(iot)_i$. The removed data can be then applied for some unapproved undertakings. The G_{way} stores the significant mystery enlistment data, which are thought to be the enrolled substances of the IoT climate.

2.3. RSM-CIoTD scheme

The proposed RSM-CIoTD scheme consists of five phases: initial formation phase, registration phase, login phase, 1st degree verification phase, and 2nd degree verification phase. Figure 1 shows the proposed RSM-CIoTD model.

2.3.1. Initial formation phase

Let g_1, g_2 are two sets of US_i of order r_2 , where r_1, r_2 are two random prime numbers. If b is a bilinear pairing of two sets g_1, g_2 , the bilinear pairing b of two sets g_1, g_2 can be denoted as: $b: g_1 \times g_1 \rightarrow g_2$. Let g_1 consists of three distinct generators ρ, ρ' and ρ'' . RA randomly chooses a 160-bit number $\kappa \in \mathbb{R}_{r_2}^*$ as its main private key. RA further selects a unique identity $ID_{S(iot)_i}$ for every $S(iot)_i$, and computes corresponding virtual identity as $VID_{S(iot)_i} = H(ID_{S(iot)_i} || \kappa_i)$. RA calculates the credential of $S(iot)_i$ as $C_{S(iot)} = H(ID_{S(iot)_i} || \kappa_{S(iot)_i} || t_{registered})$, where $t_{registered}$ is the registration timestamp (TS) of $S(iot)_i$.

Using the main private key κ , it also estimates the matching public key $K_{pub} = \kappa \cdot \rho$. Then, RA performs three hash functions for better security using SHA-256 as: $H_1: \{0,1\}^* \rightarrow \mathfrak{R}_{r_2}^*$, $H_2: \{0,1\}^* \times \{0,1\} \rightarrow \mathfrak{R}_{r_2}^*$ and $H_3: \{0,1\}^* \times g_1 \rightarrow \mathfrak{R}_{r_2}^*$. After creating all these above parameters, RA groups these parameters into a new set $S_{para} = \{r_1, r_2, g_1, g_2, K_{pub}, \rho, \rho', \rho'', H_1, H_2, H_3, VID_{S(iot)_i}, C_{S(iot)_i}\}$ and share this S_{para} to CS, US_i and $S(iot)_i$.

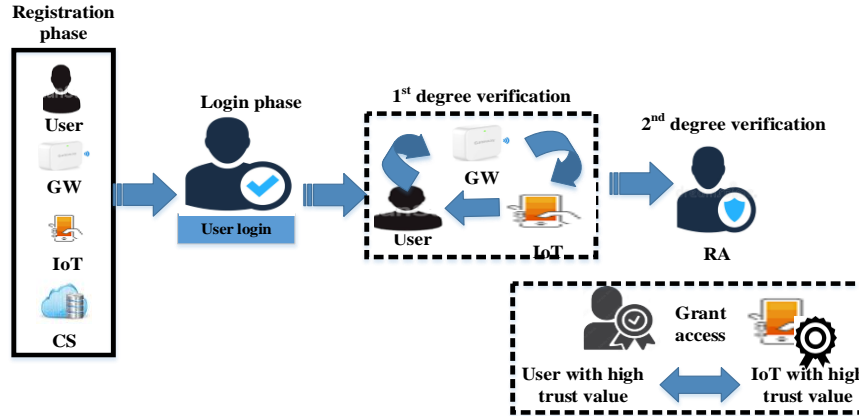


Figure 1. Proposed RSM-CIoTD model

2.3.2. Registration phase

US_i registration: to start US_i registration, US_i selects his/her identity ID_{US_i} and sends $Re\ q_{register} = R < ID_{US_i} >$ (request message for registration) to RA securely. After receiving $Re\ q_{register}$, RA generates a 160-bit main private key κ and computes virtual identity of US_i as $VID_{US_i} = H(ID_{US_i} || \kappa_i)$. In addition, RA chooses 160-bit password X and estimates $\delta_i = H(ID_{US_i} || X)$ and $\kappa'_i = H(\kappa_i || \delta_i)$. The US_i chooses a password PW_{US_i} and RA estimates a virtual password as $VPW_{US_i} = H(PW_{US_i} || X)$. Using the calculated values RA further estimates $v_{US_i} = H(VID_{US_i} || VPW_{US_i} || K_{pub}(US_i) || \delta_i)$ and store this values in its memory.

CS registration: let $CCS = \{CS_1, CS_2, \dots, CS_N\}$ be a set of legitimate CSs recorded in the network. For each $CS_i \in CCS$, the RA selects a virtual identity VID_{CS_i} . RA picks a random number $\kappa_{CS} \in \mathfrak{R}_{r_2}^*$ as the main private key of the CS. Then, it estimates $K_{pub}(CS) = \kappa_{CS} \cdot \rho$

$S(iot)_i$ registration: let a set of legitimate $S(iot)_i$ nodes $S_{IoT} = \{S(iot)_1, S(iot)_2, \dots, S(iot)_N\}$ that have been recorded in the network. For each $S(iot)_i \in S_{IoT}$, the RA chooses a virtual identity $VID_{S(iot)_i}$. Each $S(iot)_i$ maintains its own real identity $VID_{S(iot)_i}$ and $PW_{S(iot)_i}$ in its memory. RA also directs κ to the legitimate $S(iot)_i$ securely.

G_{way} registration: it is supposed that there is a secured symmetric key $SSK_{RA-G_{way}}$ among registered authority (RA) and G_{way} for their safe communication. After successful registration of US_i , RA directs the information $\{ID_{US_i}, VID_{US_i}, H(VID_{US_i} || \kappa_i)\}$ encrypted using the symmetric key $SSK_{RA-G_{way}}$, and then G_{way} store these information in its database after decryption. The G_{way} stores information $\{ID_{US_i}, VID_{US_i}, ID_i, VID_i, H(VID_{US_i} || \kappa_i), H(VID_i || \kappa_i), H(\kappa_i || \delta_i)\}$ in its database. In addition, the G_{way} also stores an order of IoT devices' virtual identities VID_i from which US_i will be able to access the real-time data.

2.3.3. Login phase

To execute the login phase, RA calculates the public key K_{pub} of US_i using the main secret key κ . RA then computes $X = r_1 \oplus H(ID_{US_i} || K_{pub}(US_i)_i^*)$, $B = r_2 \oplus H(ID_{US_i} || X)$, $VID_{US_i} = VID_{US_i}^* \oplus H(X || K_{pub}(US_i)_i^*)$ and $VPW_i^* = H(PW_{US_i}^* || X)$. After these computations, RA further calculates $v_{US_i}^* = H(VID_{US_i} || VPW_{US_i}^* || K_{pub}(US_i)_i^*)$ and then inspects the equality of $v_{US_i}^* = v_{US_i}$. If it equals, US_i offers S_{para} for verification. If it does not equal, the login phase halts instantly. Then US_i offers the current TS t_1 along with 128-bit random nonce RN_1 and RA sends a login request with 1st degree verification as $Re\ q_{1st}(Verification) = Re\ q_{US-G_{way}} = \{m1, m2, m3, m4, t_1\}$ to the G_{way} where $m1 = VID_{US_i} \oplus H(\kappa_i^* || t_1)$, $m2 = VID_{S(iot)_i} \oplus H(C_{US_i} || ID_{US_i} || t_1)$, $m3 = H(VID_{US_i} || C_{US_i} || t_1) \oplus RN_1$, $m4 = H(ID_{US_i} || VID_{S(iot)_i} || C_{US_i} || RN_1 || t_1)$, and $\kappa_i^* = \kappa_i \oplus H(ID_{US_i} || X || PW_{US_i}^* || K_{pub}(US_i)_i)$.

2.3.4. First degree verification phase

US_i to G_{way} verification: G_{way} computes $VID_{US_i} = Re\ q_{US_Gway} \oplus H(H(X||\kappa_i||t_1))$, and retrieves ID_{US_i} and C_{US_i} related to VID_{US_i} from its database. G_{way} further calculates $VID_{S(iot)_i} = m2 \oplus H(C_{US_i}||ID_{US_i}||t_1)$, and fetches $C_{S(iot)_i}$ corresponding to $VID_{S(iot)_i}$ from its database. G_{way} then computes random nonce $RN_1^* = m3 \oplus H(VID_{US_i}||C_{US_i}||t_1)$, $m5 = H(ID_{US_i}||VID_{S(iot)_i}||C_{US_i}||RN_1^*||t_1)$. G_{way} verifies the US_i to proceed further only if $m4 = m5$. Then G_{way} updates the current timestamp (TS) as t_2 and random nonce as RN_2 and sends a verification request $Re\ q_{Gway_S(iot)} = \{m6, m7, m8, t_2\}$ to the G_{way} where $m6 = H(C_{S(iot)_i}||VID_{S(iot)_i}) \oplus RN_2$, $m7 = H(VID_{US_i}||C_{US_i}||RN_1) \oplus H(C_{S(iot)_i}||t_2)$, and $m8 = H(VID_{S(iot)_i}||C_{S(iot)_i}||H(VID_{S(iot)_i}||\kappa_i||RN_2)||t_2)$.

G_{way} to $S(iot)_i$ verification: $S(iot)_i$ updates the random nonce as $RN_2^* = m6 \oplus H(C_{S(iot)_i}||VID_{S(iot)_i})$, and computes $m9 = H(VID_{S(iot)_i}||C_{S(iot)_i}||H(VID_{S(iot)_i}||\kappa_i)||RN_2^*||t_2)$. $S(iot)_i$ verifies G_{way} only if $m8 = m9$. Then $S(iot)_i$ updates the current TS as t_3 and random nonce as RN_3 and sends a verification request message $Re\ q_{S(iot)_US} = \{m10, m11, m12, t_3\}$ to US_i where $m10 = H(H(VID_{US_i}||C_{US_i}||RN_1^*||t_3) \oplus RN_3)$, $m11 = H(H(VID_{US_i}||C_{US_i}||RN_1^*||VID_{S(iot)_i}||RN_3) \oplus H(H(VID_{S(iot)_i}||\kappa_i)||t_3))$, and $m12 = H(S_{ki}||t_3)$ where S_{ki} is the session key shared with US_i denoted as $S_{ki} = H(H(VID_{US_i}||\kappa_i)||t_3||H(VID_{US_i}||C_{US_i}||RN_1^*||VID_{S(iot)_i}||RN_3)||t_3)$.

$S(iot)_i$ to US_i verification: US_i updates the random nonce $RN_3^* = m10 \oplus H(H(VID_{US_i}||C_{US_i}||RN_1^*||t_3))$, $US_i, H(H(VID_{S(iot)_i}||\kappa_i)||t_3) = m11 \oplus H(H(VID_{US_i}||C_{US_i}||RN_1^*||VID_{S(iot)_i}||RN_3))$. The US_i computes $m13 = H(S'_{ki}||t_3)$ where S'_{ki} is the session key shared with $S(iot)_i$ as $S'_{ki} = H(H(VID_{S(iot)_i}||\kappa_i)||t_3||H(VID_{US_i}||C_{US_i}||RN_1^*||VID_{S(iot)_i}||RN_3^*||t_3))$. The US_i verifies the $S(iot)_i$ only if $m12 = m13$.

Once the 1st degree verification phase succeeds, the US_i sends the request for 2nd degree verification request message to RA as $Re\ q_{2nd}\ (Verification) = \{S'_{ki}, t_4\}$. The 2nd degree verification is required to deal with a serious security threat created by an un-legitimate user/hacker who can change their behavior over time in the network. The proposed 2nd degree verification uses a set of rules created based on RSM.

2.3.5. Second degree verification phase

After receiving $Re\ q_{2nd}\ (Verification)$ from US_i , the RA verifies the timeliness of t_4 . If the condition matches, RA calculates the trust values of US_i, G_{way} and $S(iot)_i$ involved in the network communication. The trust values are calculated by a set of rules defined by RSM using three inputs namely reliability R , stability S and past score of trust T_{past} .

- Reliability R : the users involving direct communication are more reliable which can be calculated as $R = N_1(1 - DC)$ where DC is 0 or 1.
- Stability S : it postulates the time of the user endured in the same state which can be calculated as $S = N_2 \cdot stab$.
- Past score of trust T_{past} : it shows the previous trust score which can be calculated as $T_{past} = N_3 T_c$.

N_i is the normalization factor calculated as $N_i = \frac{\max(N) - N_{current}}{\max(N)}$. The use of normalization factor results the values of reliability R , stability S and past score of trust T_{past} ranging from [0 to 1]. In Table 1, T_{SR} indicates 0, T_{AVG} indicates 0.5 and T_{GR} indicates 1. Now RA computes the present trust $T_{current}$ of US_i, G_{way} and $S(iot)_i$ using Table 1. RA verifies and shares the session key S'_{ki} with US_i, G_{way} and $S(iot)_i$ only if $T_{current}(US)_i \cap T_{current}(Gway) \cap T_{current}(S(iot)_i) = T_{GR}$ and then both US_i and $S(iot)_i$ preserve the same computed session key $S_{ki} = (S'_{ki})$ for their secure communication. Otherwise, RA halts the session.

Table 1. RSM rules to calculate trust

R	S	T_{past}	$T_{current}$	R	S	T_{past}	$T_{current}$
T_{SR}	T_{AVG}	T_{SR}	T_{SR}	T_{AVG}	T_{SR}	T_{GR}	T_{GR}
T_{SR}	T_{AVG}	T_{AVG}	T_{AVG}	T_{GR}	T_{SR}	T_{SR}	T_{SR}
T_{SR}	T_{AVG}	T_{GR}	T_{GR}	T_{GR}	T_{SR}	T_{AVG}	T_{AVG}
T_{AVG}	T_{AVG}	T_{SR}	T_{AVG}	T_{GR}	T_{SR}	T_{GR}	T_{GR}
T_{AVG}	T_{AVG}	T_{AVG}	T_{AVG}	T_{SR}	T_{GR}	T_{SR}	T_{SR}
T_{AVG}	T_{AVG}	T_{GR}	T_{AVG}	T_{SR}	T_{GR}	T_{AVG}	T_{AVG}
T_{GR}	T_{AVG}	T_{SR}	T_{SR}	T_{SR}	T_{GR}	T_{GR}	T_{GR}
T_{GR}	T_{AVG}	T_{AVG}	T_{AVG}	T_{AVG}	T_{GR}	T_{SR}	T_{SR}
T_{GR}	T_{AVG}	T_{GR}	T_{GR}	T_{AVG}	T_{GR}	T_{AVG}	T_{AVG}
T_{SR}	T_{SR}	T_{SR}	T_{SR}	T_{AVG}	T_{GR}	T_{GR}	T_{GR}
T_{SR}	T_{SR}	T_{AVG}	T_{SR}	T_{GR}	T_{GR}	T_{SR}	T_{GR}
T_{SR}	T_{SR}	T_{GR}	T_{SR}	T_{GR}	T_{GR}	T_{AVG}	T_{GR}
T_{AVG}	T_{SR}	T_{SR}	T_{SR}	T_{GR}	T_{GR}	T_{GR}	T_{GR}

3. PERFORMANCE AND SECURITY ANALYSIS

3.1. Security analysis

Replay attack: The present TS concerned within the messages $m1$, $m2$, and $m13$ swapped with the communicating entities G_{way} , US_i and $S(iot)_i$. Since the messages are verified based on the received TS in these messages with a satisfactorily small proper delay t_{max} , so that A_t will not be able to replay the equal exchanged messages at some stage in login and verification phases of RSM-CIoTD scheme.

Man-in-the-middle attack: Suppose A_t snoops the login request $Re q_{1st}(Verification) = Re q_{US_Gway} = \{m1, m2, m3, m4, t_1\}$ and tries to change this message to another valid login request message. To accomplish this goal, A_t can create random nonce RN_1 and t_1 , and then try to estimate $m1 = VID_{US_i} \oplus H(\kappa_i^* || t_1)$ for altering $m1$ in $Re q_{US_Gway}$. Without knowing long term secrets VID_{US_i} , κ_i^* , and t_1 , A_t will not be able to estimate legitimate message $Re q_{US_Gway}$. In the similar way, A_t will no longer be able to create other messages which are used in the 1st and 2nd degree verification phase. Thus, RSM-CIoTD is secure against man-in-the-middle attacks.

Anonymity and untraceability: the TSs t_1 , t_2 , and t_3 and random nonces RN_1 , RN_2 , and RN_3 are used in the exchanged messages $Re q_{US_Gway}$, $Re q_{Gway_S(iot)}$, and $Re q_{S(iot)_US}$ of RSM-CIoTD throughout login and, two degree verification phases. These make the messages $Re q_{US_Gway}$, $Re q_{Gway_S(iot)}$, and $Re q_{S(iot)_US}$ separate in each phase. Thus, A_t is not able to guess US_i , G_{way} and $S(iot)_i$ during the communication.

Resilience against IoT sensor physical capture (IoT-SPC) attack: suppose N number of $S(iot)_i$ are physically captured by A_t . Let $p(N)$ signify the opportunity that A_t can decrypt secure communication between US_i and noncompromised $S(iot)_i$ after conciliating N number of $S(iot)_i$. RSM-CIoTD turn into secure against IoT-SPC attack when the rule $p(N) = 0$ is met. After the physical capturing of $S(iot)_i$, A_t can find the vital information VID_{US_i} , $VID_{S(iot)_i}$ and κ_i from its memory. It is really worth noticing that VID_{US_i} , $VID_{S(iot)_i}$ and κ_i are offered by RA and these are diverse for all $S(iot)_i$. Therefore, physical capturing of $S(iot)_i$ can only help A_t to obtain the session key between US_i and $S(iot)_i$, but not other session keys between non-compromised $S(iot)_i$ and other users. Hence, other entities in the network can nonetheless experience at ease communicate amongst them. Therefore, RSM-CIoTD is unconditionally safe towards IoT-SPC attacks.

DoS attack: in login phase of RSM-CIoTD, if a legal US_i enters his/her inappropriate ID_{US_i} and/or PW_i , it is locally proved using the condition $F_i' = F_i$. The login request of US_i is directed to the G_{way} only after successful local confirmation. When US_i tries to validate with $S(iot)_i$, his/her login request is first validated locally. Hence, RSM-CIoTD offers defense against such DoS attacks.

Session key security: Let A_t snoops $Re q_{US_Gway}$, $Re q_{Gway_S(iot)}$ and $Re q_{S(iot)_US}$ at the time of the login and two-degree verification phases of RSM-CIoTD. Without having the $Re q_{S(iot)_US}$ and virtual identities, A_t cannot preserve the secret key.

Performance analysis and comparison: communication overhead is the main component in assessing the scheme's performance. To verify a user the RA has to perform two-degree verification before data sharing in the proposed RSM-CIoTD scheme. The communication costs among RSM-CIoTD and other schemes are compared in Table 2.

Table 2. Communication cost comparison

	Communication Cost (in bits)			
	[23]	[24]	[25]	RSM-CIoTD
Total No. of bits	2752	2528	1696	1536

We consider that the identity, random nonce, hash digest (if secure hash algorithm (SHA-1) is applied) and TS are 160 bits, 128 bits, 160 bits, and 32 bits, respectively. In RSM-CIoTD, the messages $m1, m2, \dots, m13$ needs a total of $(512+512+512)=1536$ bits. It is evident from Table 2 that RSM-CIoTD performs better in terms of communication cost.

The computation cost of the proposed and existing schemes is computed based on the cryptographic operations used which is presented in Table 3. The proposed RSM-CIoTD scheme uses elliptic curve multiplication, bilinear pairing, hashing functions during login phase, 1st degree verification phase, and 2nd degree verification phase. This requires a computation cost of 62.989 ms. Similarly, the computation cost of the proposed RSM-CIoTD scheme also depends on the message exchange entities involved during communication. The different message exchange entities such as US_i , G_{way} and $S(iot)$ use different computation costs which are presented in Table 4.

Table 3. Computation cost in terms of cryptographic function used

Scheme	Cryptographic operations used	Computation cost (ms)
[23]	Elliptic curve addition, Elliptic curve multiplication, Exponential, Hashing	90.292
[24]	Elliptic curve multiplication, Bilinear pairing	103.682
[25]	Elliptic curve addition, Elliptic curve multiplication, Hashing	47.189
RSM-CIoTD	Elliptic curve multiplication, Bilinear pairing, Hashing	62.989

Table 4. Computation cost in terms of message exchange entities

Methods	Computation Cost (in ms)			
	US_i	G_{way}	$S_{(iot)}$	Total
[23]	3.52	4.48	2.24	10.24
[24]	104.20	86.78	69.36	260.34
[25]	22.54	2.56	2.88	27.98
RSM-CIoTD	5.32	4.22	4.21	13.75

Table 4 is evidence that the proposed RSM-CIoTD scheme requires more computation cost as compared to the method proposed in [25]. However, the computation cost needed for $S(iot)_i$ in the proposed RSM-CIoTD scheme almost remains the same as that for the method in [25]. This is also justified as the proposed RSM-CIoTD scheme sustains extra security and functionality features as compared to those for the method in [25].

4. CONCLUSION

In this paper, we suggest a new authentication scheme with rough set of rules for cloud IoTs. Our proposed RSM-CIoTD scheme adopts a two-degree verification scheme to make the scheme strongly secure. Two-degree verification withstands the non-public statistics disclosure attack successfully and much greater adequately. Our proposed RSM-CIoTD scheme depends on a rough set of rules and requires the secret key of the authorized user and is powerful to resist tracing attacks and insider user attacks. To provide safety in cloud IoT the proposed RSM-CIoTD scheme makes use of minimal and maximum trust upsides of past communication. A trusted RA is introduced in the proposed RSM-CIoTD scheme to affirm the cloud entities. From the comprehensive security and performance analysis, we conclude that RSM-CIoTD scheme can negotiate the secret key confidentially, resist various attacks, and consume much less strength. Further, a comparison of the proposed scheme with some existing schemes indicates its cost efficiency concerning communicate and computation costs. In future work, we can continue to analyze and implement the authentication scheme for cloud IoT and try to adopt blockchain techniques for more protection.

ACKNOWLEDGEMENTS

Registration number: 17221282162017.




REFERENCES

- [1] Z. Xia, N. N. Xiong, A. V. Vasilakos, and X. Sun, "EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud computing," *Information Sciences*, vol. 387, pp. 195–204, May 2017, doi: 10.1016/j.ins.2016.12.030.
- [2] S. K. Pasupuleti, S. Ramalingam, and R. Buyya, "An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing," *Journal of Network and Computer Applications*, vol. 64, pp. 12–22, Apr. 2016, doi: 10.1016/j.jnca.2015.11.023.
- [3] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340–352, Feb. 2016, doi: 10.1109/TPDS.2015.2401003.
- [4] W. Song, B. Wang, Q. Wang, Z. Peng, W. Lou, and Y. Cui, "A privacy-preserved full-text retrieval algorithm over encrypted data for cloud storage applications," *Journal of Parallel and Distributed Computing*, vol. 99, pp. 14–27, Jan. 2017, doi: 10.1016/j.jpdc.2016.05.017.
- [5] H. Zhu, R. Lu, C. Huang, L. Chen, and H. Li, "An efficient privacy-preserving location-based services query scheme in outsourced cloud," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7729–7739, Sep. 2016, doi: 10.1109/TVT.2015.2499791.
- [6] J. Shao, R. Lu, and X. Lin, "FINE: A fine-grained privacy-preserving location-based service framework for mobile devices," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, Apr. 2014, pp. 244–252, doi: 10.1109/INFOCOM.2014.6847945.
- [7] C. Lyu, S.-F. Sun, Y. Zhang, A. Pande, H. Lu, and D. Gu, "Privacy-preserving data sharing scheme over cloud for social applications," *Journal of Network and Computer Applications*, vol. 74, pp. 44–55, Oct. 2016, doi: 10.1016/j.jnca.2016.08.006.
- [8] J.-S. Lee, C.-J. Chew, J.-Y. Liu, Y.-C. Chen, and K.-Y. Tsai, "Medical blockchain: Data sharing and privacy preserving of EHR based on smart contract," *Journal of Information Security and Applications*, vol. 65, Art. no. 103117, Mar. 2022, doi:




- 10.1016/j.jisa.2022.103117.
- [9] A. K. G. and S. C. P., "An extensive research survey on data integrity and deduplication towards privacy in cloud storage," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 2, pp. 2011–2022, Apr. 2020, doi: 10.11591/ijece.v10i2.pp2011-2022.
- [10] M. A. D. Shewale, "IOT based smart and secure health care system analysis & Data comparison," *International Journal for Research in Applied Science and Engineering Technology*, vol. 8, no. 1, pp. 394–398, Jan. 2020, doi: 10.22214/ijraset.2020.1073.
- [11] S. S. Ambarkar and N. Shekokar, "Toward smart and secure IoT based healthcare system," in *Internet of Things, Smart Computing and Technology: A Roadmap Ahead*, 2020, pp. 283–303.
- [12] B. Farahani, M. Barzegari, F. Shams Aliee, and K. A. Shaik, "Towards collaborative intelligent IoT eHealth: From device to fog, and cloud," *Microprocessors and Microsystems*, vol. 72, p. 102938, Feb. 2020, doi: 10.1016/j.micpro.2019.102938.
- [13] R. R. Devi and V. V. Chamundeswari, "Triple DES: Privacy preserving in big data healthcare," *International Journal of Parallel Programming*, vol. 48, no. 3, pp. 515–533, Jun. 2020, doi: 10.1007/s10766-018-0592-8.
- [14] P. Nayak, S. K. Mohapatra, and S. C. M. Sharma, "Privacy and security issues in IoT cloud convergence of smart health care," in *Connected e-Health*, 2022, pp. 439–455.
- [15] Y. Zhang, M. Qiu, C.-W. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: Healthcare cyber-physical system assisted by cloud and big data," *IEEE Systems Journal*, vol. 11, no. 1, pp. 88–95, Mar. 2017, doi: 10.1109/JSYST.2015.2460747.
- [16] Y. H. Robinson, X. A. Presskila, and T. S. Lawrence, "Utilization of internet of things in health care information system," in *Internet of things and big data applications*, 2020, pp. 35–46.
- [17] S. Ben Othman, F. A. Almalki, and H. Sakli, "Internet of things in the healthcare applications: overview of security and privacy issues," in *Intelligent Healthcare*, Singapore: Springer Nature Singapore, 2022, pp. 195–213.
- [18] M. A. D. Shewale, "IOT & Raspberry Pi based smart and secure health care system using BSN," *International Journal for Research in Applied Science and Engineering Technology*, vol. 8, no. 2, pp. 506–510, Feb. 2020, doi: 10.22214/ijraset.2020.2077.
- [19] H. Kaur, M. Atif, and R. Chauhan, "An internet of healthcare things (IoHT)-based healthcare monitoring system," in *Advances in Intelligent Computing and Communication*, 2020, pp. 475–482.
- [20] A. Ullah, J. Li, A. Hussain, and E. Yang, "Towards a biologically inspired soft switching approach for cloud resource provisioning," *Cognitive Computation*, vol. 8, no. 5, pp. 992–1005, Oct. 2016, doi: 10.1007/s12559-016-9391-y.
- [21] A. M. Shabut, K. P. Dahal, S. K. Bista, and I. U. Awan, "Recommendation based trust model with an effective defence scheme for MANETs," *IEEE Transactions on Mobile Computing*, vol. 14, no. 10, pp. 2101–2115, Oct. 2015, doi: 10.1109/TMC.2014.2374154.
- [22] W. Fang, C. Zhang, Z. Shi, Q. Zhao, and L. Shan, "BTRES: Beta-based trust and reputation evaluation system for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 59, pp. 88–94, Jan. 2016, doi: 10.1016/j.jnca.2015.06.013.
- [23] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment," *Ad Hoc Networks*, vol. 36, pp. 152–176, Jan. 2016, doi: 10.1016/j.adhoc.2015.05.014.
- [24] S. Challa *et al.*, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017, doi: 10.1109/ACCESS.2017.2676119.
- [25] M. Wazid, A. K. Das, V. Bhat K, and A. V. Vasilakos, "LAM-CIoT: Lightweight authentication mechanism in cloud-based IoT environment," *Journal of Network and Computer Applications*, vol. 150, Art. no. 102496, Jan. 2020, doi: 10.1016/j.jnca.2019.102496.

BIOGRAPHIES OF AUTHORS



Sheeba MaryJohn Rukmony    obtained her bachelor's degree in computer science from St. Mary's College, Tuticorin. Then she obtained her master's degree in computer applications from V. V. Vannaiaperumal College, Virudhunagar. She has also obtained a master's degree in computer science and engineering from Rajalakshmi Engineering College, Chennai. Currently, she is doing her research at St. Xavier's College, Palayamkottai. Her current research interests are grid computing, cloud computing, and the Internet of things. She can be contacted at email: sheebasjustus@gmail.com.



Suganthi Gnanamony    received her B.Sc. degree and M.Sc. degree from the Nesanomy Memorial Christian College, Marthandam, and her Ph.D. degree from the Manonmaniam Sundaranar University, Tirunelveli. She is working as an associate professor in the Department of Computer Science, Women's Christian College, Nagercoil. She is guiding six Ph.D. scholars. She has presented 15 papers in national and international conferences and published 12 papers in international journals. She has authored 3 books. She received awards namely Shiksha Rattan Pureskar in October 2012 in New Delhi and Best Citizen Award by International Publishing House, New Delhi in February 2013. She has also received the Research Leadership Award 2020 for International Innovation, Betterment and Excellence in Research Activities. She can be contacted at email: dr_suganthi_wcc@yahoo.co.in.