IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 11, 2018, accepted November 7, 2018, date of publication December 3, 2018,
date of current version January 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2884541

Registration Center Based User Authentication Scheme for Smart E-Governance Applications in Smart Cities

SAUD S. ALOTAIBI
Department of Information Systems, College of Computer and Information Systems, Umm Al Qura University, Makkah 24381, Saudi Arabia
e-mail: ssotaibi@uqu.edu.sa

ABSTRACT Smart cities and E-governance in smart cities has been the emerging topic in this twenty-first century. The rapid developments and experiences in advancing and utilizing the applications of smart city for the continuous smart E-governance have evolved as an interesting topic to concentrate. According to Information and Communication Technology (ICT), the aims and paradigm of the smart city are to provide an integrated infrastructure and manage proper E-governance with a mission of next-generation cities. The smart delivery services which are opting the ICT-based E-governance are facilitated with an interactivity, say Internet through which the citizens can enjoy the best facilities of smart city. As the growing popularity of smart city and its applications, the communication between the government and the citizens is also raising the possibility of infringement attempts. To address this issue, we come up with an advanced multi-factor user authentication scheme which can be utilized for the smart E-governance applications in smart cities. The lightweight nature of our scheme and resistance to many network attacks, including low computational overhead show beyond doubt that our scheme is efficient and applicable to E-governance applications in smart city. A formal verification performed using AVISPA tool confirms the security of the proposed scheme.

INDEX TERMS E-governance, smart city, privacy, security, authentication, AVISPA.

I. INTRODUCTION

The delivery of information and government services over the internet by means of electronic system is defined as electronic governance or E-governance system. This type of providing the service to the information is also be pertained as information technology (IT). Using the IT, the data can be disseminated to the public and other agencies with an efficiency, faster facilities to perform the government administration with good governance. It is observed that facilities and advantages of E-governance is beyond the scope of e-government. E-governance is not just about the government websites and e-mail services. The utility of E-governance helps in bringing the change as ''how the citizens connect to governments in a way to relate themselves to each other''. This adaptation of E-governance gives freedom to develop the new concept such as citizenship, in terms of needs and responsibilities. This E-governance gives an ample advantage to the citizens to understand the government policies and participate in the governments policy-making by communicating with each other.

In the smart city E-governance, Internet of Things is also playing a vital role in providing a better services to the citizens in utilizing ICT features. Information and communication technologies can play a central role in sharing the data to millions of small devices over the Internet. This brings the security concerns in communications and information exchange.

Despite the term Smart City is very common in everyday speaking, its exact definition is still not well-established [13]. In literature, some interesting definitions can be found, as:

- ''A city connecting the physical infrastructure, the ICT infrastructure, the social infrastructure, and the business infrastructure to leverage the collective intelligence of the city'' [17].
- ''A city that invests in human and social capital and traditional and modern (ICT) communication infrastructure in order to sustain the economic growth and a high quality of life, with a wise management of natural resources, through participatory governance'' [8].
- ''A city whose community has learned to learn, adapt and innovate. People need to be able to use the technology in order to benefit from it'' [14].
- ''A city that reflects a particular idea of local community, one where city governments, enterprises and residents

use ICTs to reinvent and reinforce the community's role in the new service economy, create jobs locally and improve the quality of community life'' [4].

II. SECURITY RISKS IN E-GOVERNANCE

The effective E-governance is the important aspect in governing the smartcity. ICT is one the facilitator for the development of effective E-governance. One of the applications of ICT is E-governance which involve in delivering public services, transmitting the information over exchanging the transactions. The utility services which ICT provides can be featured as government-to-business (G2B), government-to-customer (G2C), and government-to-government (G2G). Furthermore, ICT can also interact within the framework of government. Figure 1 shows common risks in E-governance. Enriching the information security is an important and crucial aspect which is causing obstructions in effective implementation of E-governance. Information transmitted must be secured from unauthorized access for effective implementation of E-governance.

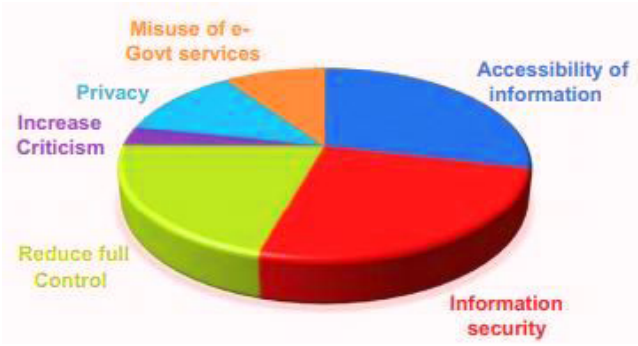

FIGURE 1. Risks in E-governance.

In getting access to the government services an attacker may breach the security check as follows:

- An attacker may create fake identities to fool the E-governance system and gain access. And once the attacker can breach the security, he/she can easily misuse the services of the server as per their desire.
- An attacker may keep the E-governance system busy and make the services unavailable to the real users by delaying or terminating the services of a host connected to the internet.
- As described by Doley and Yao method [16], an attacker can capture the transmitted messages and insert, delete, modify or resend the messages. Also, attacker can eavesdrop on the transmitted messages between the parties over the public channel.

III. RELATED WORK

In the process of tackling the attacks and security threats, in the recent years many proposals and approaches [13], [18], [23], [31]–[33] for E-governance has been seen. In the literature, there were schemes related to different remote user authentication proposals using the smartcards were suggested.

Very recently in 2015, Srinivas *et al.* [19] in his work in the design of multi-server environment able to find Shunmuganathan *et al.* [39] scheme which has many vulnerabilities to replay attack, impersonation attack, forward secrecy attack and smart card attack. Kalra and Sood [22] in his proposal which deals with the cloud servers and Internet of Things(IoT) have come up with a secure authentication scheme which involving Elliptic Curve Cryptography (ECC). But the main problem with the design is the message size which is employed during the communication is significantly increased. However, due to the provision of producing secure mutual authentication and also issues in establishing the valid session key their scheme considered insecure. In 2016, Sharma and Kalra [35], [38] come up with a proposal regarding the authentication schemes by employing quantum identity to ensure the authenticity of the user to the cloud server. To overcome the password guessing attack, many authors employ biometrics as a factor to involve in the login credentials. In developing a strong user authentication scheme one can use a scalable factor such as biometric which cannot be guessed and also possess an uniqueness property. Wazid *et al.* [46] proposed a scheme to overcome the weaknesses of the existing schemes in the literature and claims their scheme supports the functionality features (see Table.5).

In 2017, Moon *et al.* [29] designed a remote user authentication scheme to ensure the data security. Their scheme too uses the widely considered smart card which can be adopted as it has low computational cost and expedient portability. Sharma *et al.* [36] proposed a protocol which has the lightweight computations and, they claim it is a robust remote UAKA protocol for e-governance applications which can be utilized in the smart cities environment. As the E-governance has become an essential phenomenon for the smarter way of doing administration by the smart-city authority. As the transmission of sensitive and private information between the government and the user(citizen) over the internet, technically the E-governance need to improvise the information security.

In 2018, for the E-governance applications in the environment related to smart cities, Sharma *et al.* [37] presented a novel remote user scheme. Though their scheme is designed with lightweight parameters to ensure low-computational overhead, their scheme has the possibility to improvise to transmit the low bandwidth and reduce the computation time during the login and authentication phases. Ali and Pal [3] proposed an efficient three-factor-based authentication scheme for the multi-server environment which uses elliptic curve cryptography techniques. But, the problem with their scheme is high computation cost in comparison to the lightweight techniques applied by Moon *et al.* [29] and Sharma *et al.* [36], [37]. Thus, we focus on designing a lightweight user authentication scheme which can ensure the low computation cost and also successfully ensures the security features as mentioned in Table.5.

A. CONTRIBUTIONS

The major contributions of this work are listed as follows:

- In the E-governance (C2G) environment, to ensure the secure transactions among the participants, we propose a novel design to authenticate the user efficiently by issuing smartcard to the registered users.
- Our scheme is lightweight in terms of computations involved and suits the E-governance environment.
- Using ROR model, an analysis using formal security method is done which ensures that the security attacks arose by an attacker is well taken care by our scheme.
- Also, in continuation to the formal security analysis, we have presented BAN-Logic model which ensures in taking care of the freshness of the current communicated messages and also ensures the validity of the session key.
- Using a world-wide accepted formal verification tool such as AVISPA simulation tool is applied to our scheme to ensure about the interception attacks such as man-in-the middle attack and replay attack. To strengthen the security analysis on our proposed scheme an informal security analysis is carried out to overcome other uncovered attacks.
- Finally, our proposed protocol takes shape as a well defined and comparable to show its performance while comparing with the other related schemes is demonstrated to prove the novality and efficiency.

IV. MATHEMATICAL PRELIMINARIES

A. BIOHASHING

"Hash key" is a seed representation of the input from the biometric feature set. The function of this BioHashing technique is to generate a vector of bits [20], [42]. Therefore, the job of this procedure is to diminish the scope and probability of denial of access while ensuring the false acceptance performance. To be more precise, the uniformly distributed pseudo-random numbers are generated with the help of the secret seed which was inputed as the biometric vector data $x \in R^n$ is foreshortened to a bit vector $b \in \{0,1\}^l$ with the length of the bit string as l such that $(l \leq n)$ through BioHashing.

B. NOTATIONS AND RULES OF BAN LOGIC

It is considered to use BAN logic [7] to validate the correctness of a well designed AKA protocol. In this method of BAN logic the participants involved in the communications under go the verification process so that the transmitted messages need to be validated to check the interception attack. So, here the transmitted messages from a legitimate user U_i and an opted cloud server CS_j commit on freshly established shared session key which the both parties can possess. This process takes place during the execution of the protocol and hence the validation of the participants are done on the fly. For the readily use and understanding we provide Table. 1 which possess the notations utilized in defining and presenting the BAN logic.

TABLE 1. Notations of BAN-logic.

Symbol		Description
$A \equiv \alpha$	:	The principal A believes the arrived α.
$A \triangleleft \alpha$	:	A obeys α contained in a message.
$A \sim \alpha$	:	A mentioned α, as $A \equiv \alpha$ arrival
$A \Rightarrow \alpha$	:	A has complete command on α and considers trusted
$\sharp(\alpha)$	:	The message α is fresh
$A \equiv B \overset{SK}{\longleftrightarrow} A$	:	A and B use SK (shared session key).
$A \overset{SK}{\longleftrightarrow} B$	:	Shared secret between A and B.
$< \alpha >_Y$	:	α is combined with the formula Y.
(α)	:	α is hashed value.
(α, Y)	:	α and Y are combined and hashed .
$(\alpha, Y)_k$	:	α and Y are combined with hashed key k.

TABLE 2. Rules of BAN-logic.

Rules		Functionality Description
Message meaning	:	$\frac{M \equiv N \overset{k}{\longleftrightarrow} M, M \triangleleft \{X\}_k}{M \equiv N \mid \sim X}$.
Nonce verification	:	$\frac{M \equiv \sharp(X), M \equiv N \sim X}{M \equiv N \equiv X}$.
Jurisdiction	:	$\frac{M \equiv N \equiv X, M \equiv N \Rightarrow X}{M \equiv X}$.
Freshness	:	$\frac{M \equiv \sharp(X)}{M \equiv \sharp(X,Y)}$.
Belief	:	$\frac{M \equiv N \equiv (X,Y)}{M \equiv N \equiv (X)}$.
Session key	:	$\frac{M \equiv \sharp(X), M \equiv N \equiv X}{M \equiv M \overset{k}{\longleftrightarrow} N}$.

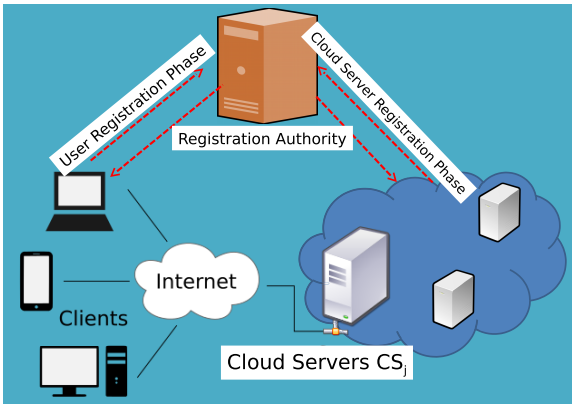

FIGURE 2. Network model.

We present the rules of BAN-Logic in Table. 2, in order to present the logical posits in the formal terms [7], [43]:

C. NETWORK MODEL

As shown in the Figure 2, the proposed protocol considers multi-cloud-server environment as the network model where the participants in this model would be the users U_i, the registration center RC, and the cloud servers CS_j as the involved entities. In this model, the RC is considered as a trusted authority and provides the system parameters to the participants during the registration phase. The registered users receives a valid smart card accumulated with some parameters as confidential values needed to login into the system and contact the registered cloud-server to avail the services. Also the cloud-servers are loaded with a secret value

which is specific to the CS_j. U_i and CS_j authenticate each other by proving the legitimacy of each other and then without the involvement or assistance of *RC* both the entities establish a secret session key. Using this session key U_i can access the desired services from CS_j.

D. ADVERSARY/THREAT MODEL

This section deals with the necessary characteristics and assumptions, including the attacker's capabilities in E-governance environment.

(1) The participants do their entire communication over the insecure channel, an attacker possess the capability to intercept or modify any messages that are transmitted among the parties over public channel [11], [43].
(2) The transmitted messages can be eavesdropped by an attacker [12], [26].
(3) By applying the power consumption analysis on the captured smartcards, an attacker can extract the valuable information stored on the smartcard [24], [28], [42].
(4) In the network, the registration authority is considered to be secure, however due to the hostile environment the deployed servers can be physically captured by an attacker [15].
(5) Using an off-line manner, an attacker can guess the low-entropy passwords and identities used by an registered user [19], [27].

V. PROPOSED SCHEME

We have come up with a scheme which presents its novality in related to applications of smart cities based on E-governance where the remote user can access the resources. Also, we have made use of lightweight parameters to specifically ensure the low-computational overhead, during the execution of login phase, and session key establishment during authentication phases.

In our scheme, the recipients and CS_j's undergo the registration process with the Cloud Service Provider (CSP) in order to avail the access of any desired information from the opted cloud server. The necessary system parameters are generated by CSP. s is the master secret key used by CSP and makes the hash functions such as $h(\cdot)$ and $H(\cdot)$ as publicly used parameters.

In many recent proposals, people make user of the current system timestamps. Therefore, we too have applied the timestamps concept in our proposed scheme to handle the replay attack vulnerability. Therefore, all the participants in the architecture of the network model (e.g., users, CSP, and CS_j) need to avail the clock-synchronization feature. In a way, this assumption is considered to be practical and much widely-considered in many designs which are recently proposed [9], [34], [40], [41]. In Table. 3, we have listed all the possible notations and their descriptions used throughout the paper.

The proposed scheme is mainly partitioned into five phases:

TABLE 3. Notations and their meanings.

Symbol	Description
U_i	i^{th} remote user
CS_j	communicated cloud-server
RA	Registration Authority in E-governance
SC_i	Smartcard of user U_i
ID_i, PW_i	Identity and Password of user U_i
ID_{CS_j}	Identity of cloud-server CS_j
$\mathcal{A}$	Attacker/Adversary
$h(\cdot)$	Collision resistant one-way hash function
ΔT	Time interval for the allowed transmission delay
T_c	Time when a message received by an entity
SK	Session key shared between U_i and CS_j
$\gamma \| \beta$	Concatenation of data γ with data β
$\gamma \oplus \beta$	Exclusive-OR of data γ and data β

a. The Registration
b. Login and Authentication
c. Password change
d. User revocation
e. Dynamic CSP addition

A. THE REGISTRATION PHASE

In our scheme, this whole process encounters in off-line mode. During this process, a genuine user produces his registration parameters over the secure channel to the registration center (*RC*) which is considered to be trusted. Thus, the communication encounters is assumed to be secure during this phase. Furthermore, looking into the practical scenarios, any registered genuine user before getting the credentials, first submits the required credential details physically, and a successful registration of the user happens only after the verification of the produced details. This phase undergoes two sub-phases namely, (i) cloud server registration, and (ii) user registration.

1) CLOUD SERVER (CS_j) REGISTRATION

The following steps are performed by *RC* in the off-line mode:

C1: CS_j being the current cloud server chooses identity ID_{CS_j} which is expected to be unique. As mentioned CS_j communicates using the secure channel to send the request to *RC* for the registration as follows:

$$CS_j \rightarrow RC : \{ID_{CS_j}\} \tag{1}$$

C2: Once the request for registration is received, for the unique $h(s\|ID_{CS_j})$ RC checks in Rec_{CS}. If the server is new, *RC* computes $C_j = h(ID_{CD_j}\|s)$ and sends it to CS_j using $IKEV_2$.

$$RC \rightarrow CS_j : \{C_j\} \tag{2}$$

and further stores $h(s\|ID_{CS_j})$ in Rec_{CS} for future verifications.

C3: On receiving the response, CS_j keeps C_j as secret.

Cloud Server (CS_j)	Registration Center (RC)
Choose ID_{CS_j}.	For unique $h(s\|\|ID_{CS_j})$ in Rec_{CS}
$\xrightarrow[Secure\ \ channel]{Reg-ReqCS_j=\{ID_{CS_j}\}}$	Compute $C_j = h(ID_{CD_j}\|\|s)$
keep C_j secret	$\xleftarrow[Secure\ \ channel]{\{C_j\}usingIKEV_2}$
	store $h(s\|\|ID_{CS_j})$ in Rec_{CS}
User U_i	**Registration Center (RC)**
Choose ID_i, PW_i, BIO_i.	
Generate random numbers r_i.	
Compute	
$NPW_i = H(BIO_i\|\|PW_i) \oplus r_i$.	For unique $h(s\|\|ID_i)$ in Rec_U
$\xrightarrow[Secure\ \ channel]{Reg-ReqU_i=\{ID_i,NPW_i\}}$	Compute $N_i = h(ID_i\|\|s) \oplus NPW_i$, $A_i = NPW_i \oplus C_j$.
compute	
$Z_i = h(ID_i\|\|A_i\|\|PW_i\|\|r_i)$,	$\xleftarrow[Secure\ \ channel]{SC_i=\{A_i,N_i,h(\cdot),H(\cdot)\}}$
$L_i = r_i \oplus h(ID_i\|\|PW_i)$,	
$B_i = A_i \oplus h(PW_i\|\|ID_i)$.	store $h(s\|\|ID_i)$ in Rec_U
$SC_i = \{B_i, L_i, Z_i, N_i, h(\cdot), H(\cdot)\}$	

FIGURE 3. Summary of registration phase.

2) USER REGISTRATION

The following steps are executed by RC for each user U_i using off-line mode :

Step 1: A user U_i chooses an identity ID_i, password PW_i and also imprints BIO_i and generates a random number r_i, and computes $NPW_i = H(BIO_i\|\|PW_i) \oplus r_i$. Then, sends a request as

$$U_i \rightarrow RC : \{ID_i, NPW_i\} \tag{3}$$

Step 2: RC checks for the uniqueness of the user in Rec_U as $h(s\|\|ID_i)$. For new user, RC computes as

$$\begin{aligned} N_i &= h(ID_i\|\|s) \oplus NPW_i \\ A_i &= NPW_i \oplus C_j \end{aligned} \tag{4}$$

Then RC sends the smartcard (SC_i) to the U_i over the secure channel as

$$RC \rightarrow U_i : SC_i = \{A_i, N_i, h(\cdot), H(\cdot)\} \tag{5}$$

Step 3: On receiving the response, the user undergoes the following computations:

$$\begin{aligned} Z_i &= h(ID_i\|\|A_i\|\|PW_i\|\|r_i) \\ L_i &= r_i \oplus h(ID_i\|\|PW_i) \\ B_i &= A_i \oplus h(PW_i\|\|ID_i) \end{aligned} \tag{6}$$

Stores Z_i, L_i and replaces A_i with B_i on the smart card. Finally, the smart card contains:

$$U_i \rightarrow SC_i : \{B_i, L_i, Z_i, N_i, h(\cdot), H(\cdot)\} \tag{7}$$

B. LOGIN AND AUTHENTICATION PHASE

Here, the recipient make use of his/her login credentials to access the cloud server by login into the system. The details are discussed as follows:

1) LOGIN

To access the information, the user produces his/her login credentials into the smart card reader such as ID_i and PW_i also imprint BIO_i.

Step 1: The smart card calculates

$$\begin{aligned} r_i &= L_i \oplus h(ID_i\|PW_i) \\ A_i &= B_i \oplus h(PW_i\|ID_i) \\ NPW_i &= r_i \oplus H(BIO_i\|PW_i) \\ verify\ \ Z_i &\stackrel{?}{=} h(ID_i\|A_i\|PW_i\|r_i) \end{aligned} \tag{8}$$

If the verification is successful, the next steps can be processed.

Step 2: The user generates a random number n_1, and undergoes the following computations

$$\begin{aligned} IDS_i &= N_i \oplus NPW_i \\ P_i &= h((A_i \oplus NPW_i) \oplus T_1) \oplus ID_i \\ K &= h(IDS_i\|(A_i \oplus NPW_i)\|T_1) \\ M_1 &= h((A_i \oplus NPW_i)\|IDS_i\|T_1) \oplus n_1 \\ Y_i &= h(ID_i\|n_1\|K\|T_1) \end{aligned} \tag{9}$$

After the computations, the smart card transmit the message to CS_j over the public channel as

$$\xrightarrow[(U_i \rightarrow CS_j)]{MSG_1=\{P_i,M_1,Y_i,T_1\}} \tag{10}$$

2) AUTHENTICATION AND KEY AGREEMENT

CS_j checks the message with the received timestamp T_r once the login request is received to verify the replay/forgery messages.

Step 3: CS_j computes

$$\begin{aligned} ID_i &= P_i \oplus h(C_j \oplus T_1) \\ Verify &\quad h(s\|ID_i)\ in\ Rec_U \\ Compute &\quad CU_i = h(ID_i\|s) \\ K &= h(CU_i\|C_j\|T_1) \\ n_1 &= M_1 \oplus h(C_j\|CU_i\|T_1) \\ Verify\ \ Y_i &\stackrel{?}{=} h(ID_i\|n_1\|K\|T_1) \end{aligned} \tag{11}$$

up on the successful validation, proceed with the next steps. Else, terminates the process.

Step 4: A random number n_2 is generated by CS_j, which undergoes the following computations

$$\begin{aligned} M_2 &= n_2 \oplus K \\ SK_{ij} &= h(n_1\|CU_i\|n_2\|C_j\|T_2) \\ M_3 &= h(ID_i\|n_1\|SK_{ij}\|n_2\|T_2) \end{aligned} \tag{12}$$

After the computations, CS_j transmit the message to user(SC_i) over the public channel as

$$\xrightarrow[(CS_j \rightarrow U_i)]{MSG_2=\{M_2,M_3,T_2\}} \tag{13}$$

U_i (Smart card)	Cloud Server (CS_j)
Choose ID_i, PW_i and Imprint BIO_i.	
Compute $r_i = L_i \oplus h(ID_i \| PW_i)$	
$A_i = B_i \oplus h(PW_i \| ID_i)$	
$NPW_i = r_i \oplus H(BIO_i \| PW_i)$	
Verify $Z_i \stackrel{?}{=} h(ID_i \| A_i \| PW_i \| r_i)$	Check $\|T_r - T_1\| \leq \Delta T$
Generate a random number n_1	$ID_i = P_i \oplus h(C_j \oplus T_1)$
$IDS_i = N_i \oplus NPW_i$	Verify $h(s \| ID_i)$ in Rec_U
	Compute $CU_i = h(ID_i \| s)$
$P_i = h((A_i \oplus NPW_i) \oplus T_1) \oplus ID_i$	$K = h(CU_i \| C_j \| T_1)$
$K = h(IDS_i \| (A_i \oplus NPW_i) \| T_1)$	$n_1 = M_1 \oplus h(C_j \| CU_i \| T_1)$
$M_1 = h((A_i \oplus NPW_i) \| IDS_i \| T_1) \oplus n_1$	Verify $Y_i \stackrel{?}{=} h(ID_i \| n_1 \| K \| T_1)$
$Y_i = h(ID_i \| n_1 \| K \| T_1)$	Generate a random number n_2
$\xrightarrow[(U_i \to CS_j)]{MSG_1 = \{P_i, M_1, Y_i, T_1\}}$	Compute $M_2 = n_2 \oplus K$
	$SK_{ij} = h(n_1 \| CU_i \| n_2 \| C_j \| T_2)$
	$M_3 = h(ID_i \| n_1 \| SK_{ij} \| n_2 \| T_2)$
	$\xleftarrow[(CS_j \to U_i)]{MSG_2 = \{M_2, M_3, T_2\}}$
Check $\|T_r - T_2\| \leq \Delta T$	
$n_2 = M_2 \oplus K$	
$SK_{ij} = h(n_1 \| IDS_i \| n_2 \| C_j \| T_2)$	
Verify $M_3 \stackrel{?}{=} h(ID_i \| n_1 \| SK_{ij} \| n_2 \| T_2)$	
$M_4 = SK_{ij} \oplus K \oplus T_3$	
$\xrightarrow[(U_i \to CS_j)]{MSG_3 = \{M_4, T_3\}}$	Check $\|T_r - T_3\| \leq \Delta T$
	$M_4 \stackrel{?}{=} SK_{ij} \oplus K \oplus T_3$

FIGURE 4. Summary of login and authentication phase.

Step 5: SC_i checks the message with the received timestamp T_r to verify the replay/forgery messages upon receiving the response message. Further, undergoes the following computations

$$\begin{aligned} n_2 &= M_2 \oplus K \\ SK_{ij} &= h(n_1 \| IDS_i \| n_2 \| C_j \| T_2) \\ Verify\ \ M_3 &\stackrel{?}{=} h(ID_i \| n_1 \| SK_{ij} \| n_2 \| T_2) \\ M_4 &= SK_{ij} \oplus K \oplus T_3 \end{aligned} \tag{14}$$

After the computations, the smart card transmit the message to CS_j over the public channel as

$$\xrightarrow[(U_i \to CS_j)]{MSG_3 = \{M_4, T_3\}} \tag{15}$$

Step 6: CS_j checks the message with the received timestamp T_r to verify the replay/forgery messages.

$$Verify\ \ M_4 \stackrel{?}{=} SK_{ij} \oplus K \oplus T_3 \tag{16}$$

Once the verification is successful, both U_i and CS_j make use of the established session key in their future communications. Thus, completes the login and authentication process.

C. PASSWORD AND BIOMETRIC UPDATE PHASE

A registered user U_i executes this phase to update/modify his/her current password and biometric key to a new password and new biometric key. Thus, the user undergoes the following procedure:

Step 1: U_i chooses ID_i, PW_i and also imprints BIO_i. Then, computes

$$\begin{aligned} r_i &= L_i \oplus h(ID_i \| PW_i) \\ A_i &= B_i \oplus h(PW_i \| ID_i) \\ NPW_i &= r_i \oplus H(BIO_i \| PW_i) \\ verify\ \ Z_i &\stackrel{?}{=} h(ID_i \| A_i \| PW_i \| r_i) \end{aligned} \tag{17}$$

If the verification is successful, the next steps are executed. Otherwise, terminates the process.

Step 2: U_i chooses new password PW_i^{new} and also imprints new BIO_i^{new}. Then, the user performs the following computations:

$$\begin{aligned} A_i^{new} &= A_i \oplus NPW_i \oplus r_i \oplus H(BIO_i^{new} \| PW_i^{new}) \\ B_i^{new} &= A_i^{new} \oplus h(PW_i^{new} \| ID_i) \\ Z_i^{new} &= h(ID_i \| A_i^{new} \| PW_i^{new} \| r_i) \\ N_i^{new} &= N_i \oplus NPW_i \oplus r_i \oplus H(BIO_i^{new} \| PW_i^{new}) \\ L_i^{new} &= r_i \oplus h(ID_i \| PW_i^{new}) \end{aligned} \tag{18}$$

Step 3: U_i updates B_i, L_i, Z_i & N_i with $B_i^{new}, L_i^{new}, Z_i^{new}$ & N_i^{new}, respectively. Finally, the smart card contains:

$$U_i \to SC_i : \{B_i^{new}, L_i^{new}, Z_i^{new}, N_i^{new}, h(\cdot), H(\cdot)\} \tag{19}$$

D. SMART CARD REVOCATION PHASE

The needy of this phase is required, if the registered user's smart card is stolen/lost. This procedure is important in order to get new smart card SC_i^{new}:

Step 1: U_i keeps ID_i, but chooses PW_i^{new} and imprints BIO_i^{new} and generates a random number r_i^{new}, and computes $NPW_i^{new} = H(BIO_i^{new} \| PW_i^{new}) \oplus r_i^{new}$. Then, directs a request as

$$U_i \to RC : \{ID_i, NPW_i^{new}\} \tag{20}$$

Step 2: RC checks for the revoked user in Rec_U as $h(s \| ID_i)$. If user exists, RC computes as

$$\begin{aligned} N_i^{new} &= h(ID_i \| s) \oplus NPW_i^{new} \\ A_i^{new} &= NPW_i^{new} \oplus C_j \end{aligned} \tag{21}$$

Then RC sends the renewed smartcard (SC_i) to the U_i over the secure channel as

$$RC \to U_i : SC_i = \{A_i^{new}, N_i^{new}, h(\cdot), H(\cdot)\} \tag{22}$$

Step 3: On receiving the response, the user undergoes the following computations:

$$\begin{aligned} Z_i^{new} &= h(ID_i \| A_i^{new} \| PW_i^{new} \| r_i^{new}) \\ L_i^{new} &= r_i^{new} \oplus h(ID_i \| PW_i^{new}) \\ B_i^{new} &= A_i^{new} \oplus h(PW_i^{new} \| ID_i) \end{aligned} \tag{23}$$

Stores Z_i^{new}, L_i^{new} and replaces A_i^{new} with B_i^{new} on the smartcard. Finally, the smartcard contains:

$$U_i \to SC_i : \{B_i^{new}, L_i^{new}, Z_i^{new}, N_i^{new}, h(\cdot), H(\cdot)\} \tag{24}$$

E. DYNAMIC CLOUD SERVER ADDITION PHASE

Suppose a new cloud server wishes to be a part of the existing network, then CS_j^{new} needs to follow the steps in the off-line mode:

Step 1: CS_j^{new} chooses its unique identity $ID_{CS_j}^{new}$. Through a secure channel CS_j^{new} sends registration request to RC as follows:

$$CS_j \rightarrow RC : \{ID_{CS_j}^{new}\} \tag{25}$$

Step 2: On receiving the request, for the unique $h(s\|ID_{CS_j}^{new})$ RC checks in Rec_{CS}. If the server is new, RC computes $C_j^{new} = h(ID_{CD_j}^{new}\|s)$ and sends it to CS_j^{new} using $IKEV_2$.

$$RC \rightarrow CS_j^{new} : \{C_j^{new}\} \tag{26}$$

and further stores $h(s\|ID_{CS_j}^{new})$ in Rec_{CS} for future verifications.

Step 3: CS_j^{new} keeps C_j^{new} as secret upon getting the response.

VI. SECURITY ANALYSIS

This section deals with the investigation of the security analysis on the proposed scheme by performing the formal and informal security analysis to ensure the security against various known attacks. Recently, in an analysis presented by Wang *et al.* [45] investigated and presented the following interesting observation. The formal methods (for example, random oracle model) which is widely-applied cannot capture some structural faults. Therefore, considering the security of the authentication protocols is still remains an open issue. Thus, considering the important observation made by Wang *et al.* we also need to formulate the non-mathematical security analysis which is an informal way to prove the security of the authentication protocol. Furthermore, using AVISPA tool the formal security verification is carried out to show secure with high probability.

A. FORMAL SECURITY ANALYSIS USING ROR MODEL

This analysis is carried out based on Abdalla *et al.*'s [2] proposal, in this method, the formal security analysis validates and proves the security of the session key with the applicability of ROR model which is acceptable and many of the recent authentication protocols made use of this model [10], [34], [40]. Prior to proving the session key proof to the Theorem 1, we consider the primitives which are formulated with the ROR model.

Participants: Here, we indicate $\mathcal{CI}_{U_i}^{l_1}$, $\mathcal{CI}_{RC}^{l_2}$ and $\mathcal{CI}_{CS_j}^{l_3}$ as the l_1^{th}, l_2^{th} and l_3^{th} of U_i, RC and CS_j, are named as *oracles* [9].

Accepted state: Let the instance $\mathcal{CI}^l$ found to be in an acceptance state once it reach the final execution of the protocol message, then we consider it in an accepted state. For each present session, $\mathcal{CI}^l$ is given a session identification (sid) to construct an ordered concatenation to all its communicated messages during sent & receive.

Partnering: In this part, the partnering of each other for the two instances, namely $\mathcal{CI}^{l_1}$ and $\mathcal{CI}^{l_2}$ exist, if they successfully meet the three requirements: 1) both the instances are considered to be in accepted states, 2) both the instances are considered to be authenticated mutually, and 3) both the instances are mutual partners to each other.

Freshness: $\mathcal{CI}_{U_i}^{l_1}$ or $\mathcal{CI}_{CS_j}^{l_3}$ is considered to be fresh, if an adversary/attacker $\mathcal{A}$ do not possess the session key SK_{ij} by considering the *Reveal* query.

In this case, we consider that the adversary/attacker $\mathcal{A}$ has the full control over the public channel and the communicated messages among the entities. Therefore, $\mathcal{A}$ has the potential to intercept, modify, insert or erase the desired messages which are illegally transmitted during the communication by the entities. Furthermore, $\mathcal{A}$ undergoes the full access to the following oracles:

Execute($\mathcal{CI}_{U_i}^{l_1}, \mathcal{I}_{CS_j}^{l_3}$): During the execution of this query $\mathcal{A}$ communicates to U_i, and CS_j by intercepting th transmitted messages. $\mathcal{A}$ who can impose potentially interception of the messages by eavesdropping attack.

Reveal($\mathcal{CI}^l$): The session key SK_{ij} established between $\mathcal{CI}^l$ and its partner for the present execution is exposed to $\mathcal{A}$ on execution of this query.

Send($\mathcal{CI}^l, \mathcal{MSZ}$): This query is defined as active attack by $\mathcal{A}$. Once this query is executed, $\mathcal{A}$ can transmit a message $\mathcal{MSZ}$ to the desired participant at an instance $\mathcal{CI}^l$, and it can be replayed over and over. Furthermore, accordingly receives a message as a response in corresponding to the transmitted $\mathcal{MSZ}$.

CorruptSmartcard($\mathcal{CI}_{U_i}^{l_1}$): Through the execution of this query, the credentials stored on the lost/stolen smartcard can be extracted by $\mathcal{A}$.

Furthermore, the query *CorruptSmartcard*($\mathcal{CI}_{U_i}^{l_1}$) assure the weak-corruption model as pointed out in [9], which says that the ephemeral secrets (keys) are not corrupted for the participant instances.

Test($\mathcal{CI}^l$): In this query, $\mathcal{A}$ tries to capture the established semantic security of SK_{ij} between U_i and CS_j. An unbiased coin c is tossed before the start of the game, the output result is assumed to be known to $\mathcal{A}$ and plays a decision point of this query. Furthermore, the session key SK_{ij} remains to be fresh during the execution of this query, $\mathcal{CI}^l$ outputs SK_{ij} if the case is turnouts to be $c = 1$ else an arbitrary number is considered for case $c = 0$. Otherwise, null ($\perp$) is considered as an output.

Furthermore to this, if all participating entities including the attacker/adversary $\mathcal{A}$ will have the complete access to the collision-resistant one-way hash function $h(\cdot)$, where $h(\cdot)$ is modeled as a random oracle, say $H(\cdot)$.

Theorem 1: Assume that, the proposed scheme $\mathcal{P}$ runs beyond the polynomial time P_{rt}, then $\mathcal{A}$ running in polynomial time P_{rt} and the number of queries for *Send*, *Hash* queries, the number of bits in the biometrics key BIO_i, the space of $h(\cdot)$ is ranged, the size password which are of a uniformly distributed by dictionary D, are denoted as q_s,

q_h, l, $|H(\cdot)|$, $|D|$, and $\mathcal{A}$ has the the advantage in breaking the E-governance with in the time P_{rt} is defined as $Adv_{\mathcal{A}}^{E-gov}(P_{rt})$. Therefore, $\mathcal{A}$'s advantage in breaking the SK_{ij} security is estimated to be $Adv_{\mathcal{A}}^{\mathcal{P}}(P_{rt}) \leq \frac{q_h^2}{|H(\cdot)|} + 2\left(\frac{q_s}{2^l.|D|} + Adv_{\mathcal{A}}^{E-gov}(P_{rt})\right)$.

Proof: To prove this theorem, we consider the method of proving the proof similar to that which is given in [9], [40]. In this regard, we undergo our protocol with the five games, say $Game_j, j = 0, 1, 2, 3, 4$ are needed. Let $Succ_{Game_j}$ be denoted as the success probability of winning the $Game_j$ by the attacker $\mathcal{A}$ with the guessing of the correct bit c and considering the corresponding advantage probability of $\mathcal{A}$ is defined as $Adv_{Game_j} = Pr[Succ_{Game_j}]$. The detailed analysis of each game $Game_j$ is described as follows:

$Game_0$: In the beginning of the game $Game_0$ is the actual security experiment run against the proposed scheme $\mathcal{P}$ by the $\mathcal{A}$ and a challenger $\mathcal{C}$. $\mathcal{A}$ picks the bit c at the start of $Game_0$. Applying the semantic security definition to our scheme gives

$$Adv_{\mathcal{A}}^{\mathcal{P}}(P_{rt}) = |2Adv_{Game_0} - 1|. \tag{27}$$

$Game_1$: $Game_1$ is described to be eavesdropping attack. According to this game, $MSG_1 = \{P_i, M_1, Y_i, T_1\}$, $MSG_2 = \{M_2, M_3, T_2\}$ and $MSG_3 = \{M_4, T_3\}$ which are communicated among U_i, and CS_j during the execution of the login & authentication phases, $\mathcal{A}$ intercept the $Execute(\mathcal{CI}_{U_i}^{l_1}, \mathcal{CI}_{CS_j}^{l_3})$ query. Further, the *Test* query is executed by $\mathcal{A}$ to validate the output to ensure the session key/random value. $SK_{ij}^* = h(n_1\|h(ID_i\|s)\|n_2\|C_j\|T_2)$ $(= SK_{ij})$, where $C_j = h(ID_{CD_j}\|s)$, $n_1 = M_1 \oplus h(C_j\|h(ID_i\|s)\|T_1)$, and $n_2 = M_2 \oplus K$ is the session key. $\mathcal{A}$ tries to deduce SK_{ij} by making use of temporal secrets n_1 and n_2, and s which is a long term secret. Since interception of the messages MSG_1, MSG_2 and MSG_3 cannot be compromised. Therefore, winning the $Game_1$ does not effect all. Thus,

$$Adv_{Game_1} = Adv_{Game_0}. \tag{28}$$

$Game_2$: The difference between $Game_2$ and $Game_1$ is that the queries such as *Send* and *Hash* are used in $Game_2$. In this game, $\mathcal{A}$ perfoms active attack and tries to fabricate the messages to convince the participants. The computation of *Hash* query and verifies the collision with the MSG_1, MSG_2 and MSG_3, but due to the random nonces, timestamps, ID_i, ID_{CS_j} and long-term secret used in these messages. Thus, the *Send* queries by $\mathcal{A}$ is considered with a negligible probability of collision. The application of the birthday paradox is applied to get

$$|Adv_{Game_2} - Adv_{Game_1}| \leq q_h^2/(2|H(\cdot)|). \tag{29}$$

$Game_3$: The game $Game_2$ is converted to the game $Game_3$ by applying the query $CorruptSmartcard(\mathcal{CI}_{U_i}^{l_1})$. $\mathcal{A}$ extracts the credentials B_i, L_i, Z_i, and N_i. Now from $Z_i = h(ID_i\|A_i\|PW_i\|r_i)$, $L_i = r_i \oplus h(ID_i\|PW_i)$, $NPW_i = H(BIO_i\|PW_i) \oplus r_i$, and $B_i = A_i \oplus h(PW_i\|ID_i)$ guessing the correct ID_i and PW_i of U_i by $\mathcal{A}$. To achieve this task the secret credentials r_i and biometric key BIO_i are required. Thus, if we limit the number of incorrect identity/password or non-matching biometric input, we have:

$$|Adv_{Game_3} - Adv_{Game_2}| \leq q_s/(2^l.|D|). \tag{30}$$

$Game_4$: $\mathcal{A}$ tries to calculate SK_{ij} which is transmitted between U_i and CS_j with the intercepted MSG_1, MSG_2 and MSG_3, and also solving the E-gov. To compute the session key $SK_{ij}^* = h(n_1\|h(ID_i\|s)\|n_2\|C_j\|T_2)$ $(= SK_{ij})$, $\mathcal{A}$ needs $C_j = h(ID_{CD_j}\|s)$, $n_1 = M_1 \oplus h(C_j\|h(ID_i\|s)\|T_1)$, and $n_2 = M_2 \oplus K$. To deduce SK_{ij}, $\mathcal{A}$ have to get the temporal secrets n_1 and n_2, and long-term secret s.

On the other hands, $\mathcal{A}$ has C_j which he/she requires s to calculate $C_j = h(ID_{CD_j}\|s)$. i.e., $\mathcal{A}$ requires to solve E-gov in at most run time P_{rt} to deduce SK_{ij}. Thus,

$$|Adv_{Game_4} - Adv_{Game_3}| \leq Adv_{\mathcal{A}}^{E-gov}(P_{rt}). \tag{31}$$

Once $\mathcal{A}$ undergo all the queries required to win the game, he is left to guess the bit c by applying the *Test* query.

$$Adv_{Game_4} = 1/2. \tag{32}$$

(27), (28) and (32) give the following relation:

$$\begin{aligned} \frac{1}{2}Adv_{\mathcal{A}}^{\mathcal{P}}(P_{rt}) &= |Adv_{Game_0} - \frac{1}{2}| \\ &= |Adv_{Game_1} - Adv_{Game_4}|. \end{aligned} \tag{33}$$

Applying triangular inequality to the equations (29), (30) and (31), we have:

$$\begin{aligned} |Adv_{Game_1} - Adv_{Game_4}| &\leq |Adv_{Game_1} - Adv_{Game_2}| \\ &\quad + |Adv_{Game_2} - Adv_{Game_4}| \\ &\leq |Adv_{Game_1} - Adv_{Game_2}| \\ &\quad + |Adv_{Game_2} - Adv_{Game_3}| \\ &\quad + |Adv_{Game_3} - Adv_{Game_4}| \\ &\leq q_h^2/(2|H(\cdot)|) + q_s/(2^l.|D|) \\ &\quad + Adv_{\mathcal{A}}^{E-gov}(P_{rt}). \end{aligned} \tag{34}$$

Finally, (33) and (34) give the required result: $Adv_{\mathcal{A}}^{\mathcal{P}}(P_{rt}) \leq \frac{q_h^2}{|H(\cdot)|} + 2\left(\frac{q_s}{2^l.|D|} + Adv_{\mathcal{A}}^{E_gov}(P_{rt})\right)$. ■

B. SECURITY ANALYSIS USING BAN-LOGIC

Using this formal security analysis with the help of BAN logic, the mutual authentication and session key is defined for the proposed scheme between the participants U_i and CS_j [30].

Goals : In presenting the method to make sure about the establishment of session key a well defined goals need to be established. These goals are tested by formulating them as follows [7]:

Goal 1. $U_i|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$;

Goal 2. $U_i|\equiv CS_j|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$;

Goal 3. $CS_j|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$;

Goal 4. $CS_j|\equiv U_i|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$.

Before executing, we initially formulate the idealization of the messages which are used in communication during the execution of the proposed scheme between U_i and CS_j:

MSG 1: $U_i \rightarrow CS_j : \langle P_i, ID_i, n_1, T_1, M_1, U_i \overset{h(ID_{CS_j}\|s)}{\longleftrightarrow} CS_j, Y_i, U_i \overset{K}{\longleftrightarrow} CS_j \rangle_{U_i \overset{h(ID_i\|s)}{\longleftrightarrow} CS_j}$.

MSG 2: $CS_j \rightarrow U_i : \langle M_2, n_2, T_2, U_i \overset{K}{\longleftrightarrow} CS_j, M_3 \rangle_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

MSG 3: $U_i \rightarrow CS_j : \langle M_4, T_3, U_i \overset{K}{\longleftrightarrow} CS_j, n_1, n_2, \rangle_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

Some assumptions need to be formulated based on the proposed scheme such as:

A_1: $U_i|\equiv \sharp(n_1)$;

A_2: $CS_j|\equiv \sharp(n_2)$;

A_3: $U_i|\equiv (U_i \overset{K}{\longleftrightarrow} CS_j)$;

A_4: $CS_j|\equiv (U_i \overset{K}{\longleftrightarrow} CS_j)$;

A_5: $U_i|\equiv CS_j|\Rightarrow (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$;

A_6: $CS_j|\equiv U_i|\Rightarrow (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$.

Considering the BAN logic rules and taking the assumptions as described above into consideration, the communicated parties utilize the analysis in proving the mutual authentication and then go for the establishment of the session key as follows:

From the message 1, we can see:

Step 1: $CS_j \triangleleft (ID_i, P_i, U_i \overset{h(ID_{CS_j}\|s)}{\longleftrightarrow} CS_j, n_1, M_1, Y_i)_{U_i \overset{K}{\longleftrightarrow} CS_j}$.

Accounting the Step 1 & assumption A_3, considering the message meaning rule to derive:

Step 2: $CS_j|\equiv U_i|\sim (U_i \overset{K}{\longleftrightarrow} CS_j, n_1, U_i \overset{h(ID_{CS_j}\|s}{\longleftrightarrow} CS_j)$.

From Step 2 & A_1, the freshness conjuncatenation rule is enforced to acquire:

Step 3: $CS_j|\equiv \sharp(U_i \overset{K}{\longleftrightarrow} CS_j, n_1, U_i \overset{h(ID_{CS_j}\|s)}{\longleftrightarrow} CS_j)$.

At this stage, the nonce-verification rule is enforced on the Steps 2 and 3 and derive:

Step 4: $CS_j|\equiv U_i|\equiv (U_i \overset{K}{\longleftrightarrow} CS_j, n_1, U_i \overset{h(ID_{CS_j}\|s)}{\longleftrightarrow} CS_j)$.

Considering Step 4, the belief rule is enforced to acquire:

Step 5: $CS_j|\equiv U_i|\equiv (U_i \overset{K}{\longleftrightarrow} CS_j)$.

Choosing Step 5 & A_4, jurisdiction rule is enforced to acquire:

Step 6: $CS_j|\equiv (U_i \overset{K}{\longleftrightarrow} CS_j)$.

From the Message 2, we acquire:

Step 7: $U_i \triangleleft (n_2, U_i \overset{K}{\longleftrightarrow} CS_j, ID_{CS_j})_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

We apply the seeing rule on Step 7, and get:

Step 8: $U_i \triangleleft (M_2, M_3)$, where $M_2 = n_2 \oplus K$, $CU_i = h(ID_i\|s)$, $K = h(IDS_i \|(A_i \oplus NPW_i)\|T_1) = h(CU_i\|C_j\|T_3)$ and $M_3 = h(ID_i\|n_1\|SK_{ij}\|n_2\|T_2)$.

We utilize the message meaning rule on the Step 8 and A_5 to deduce:

Step 9: $U_i|\equiv CS_j|\sim (n_2, U_i \overset{K}{\longleftrightarrow} CS_j, ID_{CS_j})_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

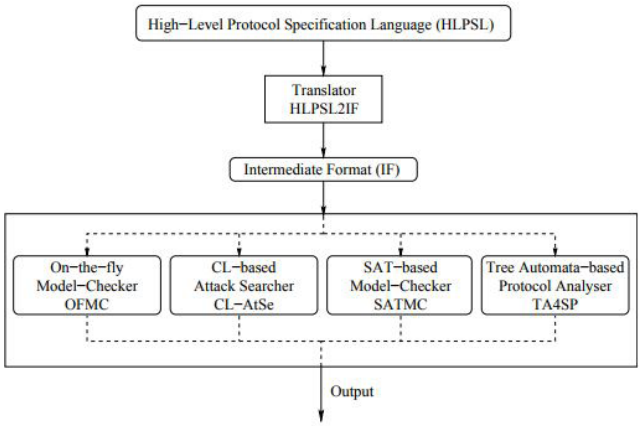

FIGURE 5. Architecture of AVISPA simulation tool.

The freshness conjuncatenation rule is applied on Step 9 & A_2, to get:

Step 10: $U_i|\equiv \sharp(n_2, U_i \overset{K}{\longleftrightarrow} CS_j, ID_{CS_j})_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

We apply the nonce-verification rule on the Steps 9 and 10, to deduce:

Step 11: $U_i|\equiv CS_j|\equiv (n_2, U_i \overset{K}{\longleftrightarrow} CS_j, ID_{CS_j})_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

Considering Step 11, belief rule is enforced to acquire:

Step 12: $U_i|\equiv CS_j|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$. (Goal 2)

Considering A_5 and the Step 12, jurisdiction rule is enforced to acquire:

Step 13: $U_i|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$. (Goal 1)

From the message 3, we infer:

Step 14: $CS_j \triangleleft (ID_i, n_1, U_i \overset{K}{\longleftrightarrow} CS_j, n_2, ID_{CS_j})_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

We consider the message meaning rule on Step 14 and assumption A_3, to deduce:

Step 15: $CS_j|\equiv U_i|\sim (n_1, n_2, ID_{CS_j}, U_i \overset{K}{\longleftrightarrow} CS_j)_{U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j}$.

According to Step 15 & A_2, the freshness conjuncatenation rule is applied to get:

Step 16: $CS_j|\equiv \sharp(n_1, n_2, ID_{CS_j}, U_i \overset{K}{\longleftrightarrow} CS_j)$.

The belief rule is applied on Step 16, to obtain:

Step 17: $CS_j|\equiv U_i|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$. (Goal 4)

Taking the assumption A_4 & Step 17, jurisdiction rule is enforced to acquire:

Step 18: $CS_j|\equiv (U_i \overset{SK_{ij}}{\longleftrightarrow} CS_j)$. (Goal 3)

In view of the Steps 12, 13, 17, and 18, the mutual authentication and key agreement is established successfully in our proposed scheme by successfully accomplishing the defined goals (Goals 1-4). Thus, the user U_i and targeted cloud server CS_j establish session key $SK_{ij} = h(n_1\|h(ID_i\|s)\|n_2\|C_j\|T_2)$ with each other which is believed to be secure.

C. AVISPA SIMULATION TOOL: FORMAL SECURITY VERIFICATION METHOD

As a part of thorough formal security presentation, we utilize the simulation method which validates the replay attack and

man-in-the-middle attack on our proposed scheme. Therefore, [5] comprises of four variant backends (Figure.5):

a. OFMC
b. CL-AtSe
c. SATMC
d. TA4SP

As shown in the architecture (Figure 5), a state-of-art-automatic analysis techniques will be carried out on the backends. As a part of the implementation we consider the role-oriented language which is popularly cognized as High-Level Protocol Specification Language(HLPSL). This HLPSL is in-built with user defined basic roles which are formulated according to the designed scheme and are executed over the network involving all the participants. Furthermore, in this HLPSL few mandatory roles exist such as session roles which are facilitated to enhance the concrete arguments which involves the entities applicable in the basic roles. However, the entire execution of the HLPSL depends on the top-level role which is the environment which can compare the sessions included in the proposed scheme involving the global constants.

The role intruder(specifically denoted as i) acts as legitimate user in the HLPSL. This gives the intruder to participate in the execution as a concrete session while executing the protocol. In [44], the detail information of AVISPA & HLPSL working validations and specifications. For the clear view of how the basic roles act during the simulation for user U_i (see Figure.6), RC (see Figure.7) and CS_j (see Figure.8), and also (see Figure.9) for the mandatory roles which are implemented specifically for the session, goal and environment is facilitated.

To simulate AVISPA, Security Protocol ANimator (SPAN) [6], presents the backends such as OFMC and CL-AtSe which are widely-accepted in many recent works. Furthermore, the other two back-ends namely SATMC and TA4SP doesn't support bitwise XOR operation. This shows that SATMC and TA4SP backends results are not conclusive. Thus, we omit the results of SATMC and TA4SP and consider only OFMC and CL-AtSe. Thus, the implementation results are reflected in the Figure.10 under OFMC as 604 nodes visited with approximated search time 29.96 seconds, and the analysis reaches the depth to 6 *plies*. From the result, it is clear that our scheme demonstrates safe and secure. Fig.10 insights us and assures the proposed scheme gratifies by securing against the man-in-the-middle attack and replay attack and satisfies the design properties.

D. INFORMAL SECURITY ANALYSIS

This section focus on the other security attacks which can be found in the network if any drawbacks exist in the proposed scheme. Well defined scheme should have the potential to restrict such attacks and threats in the network. The details are as follows:

```
%%%%%%%%%% Role for user Ui %%%%%%%%%%%%%%%%%%%%
role user(Ui, RC, CSj : agent,
% symmetric key between Ui and RC
SKuirc : symmetric_key,
% H is hash function
H : hash_func,
BI : hash_func,
SEND, RECV: channel(dy))
played_by Ui
def=
local State : nat,
IDi, IDCSj, PWi, Ri, S : text,
NPWi, BIOi, Ni, Ai, Cj, Zi, Li, Bi : text,
IDSi, Pi, K, M1, Yi, N1, N2, M2, M3, M4 : text,
SKij, T1, T2, T3, Tr : text
const user_server_n1, server_user_n2,
sub1, sub2, sub3 : protocol_id
init State := 0
transition
% User registration phase
% Ui sends < IDi, NPWi > to RC via a secure channel
1. State = 0 /\ RECV(start) =|>
State' := 2 /\ SEND({IDi. xor(Ri,H(BIOi.PWi))}_SKuirc)
/\ secret({PWi,S}, sub1, Ui)
/\ secret({IDi,IDCSj}, sub2, {Ui,RC,CSj})
% Ui receives <smart card> from RC via a secure channel
2. State = 2 /\ RECV({xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)).
xor(xor(Ri,H(BIOi.PWi)),H(IDi.S)).H.BI}_SKuirc) =|>
% Login phase
% Ui sends < Pi, M1, Yi, T1 > to CSj via a public channel
State' := 4 /\ secret({S}, sub3, RC) /\ N1' := new()
/\ T1' := new() /\ T3' := new() /\ N2' := new()
/\ IDSi' := xor(N1',xor(Ri,H(BIOi.PWi)))
/\ Pi' := xor(H(xor(xor(xor(Ri,H(BIOi.PWi)),
H(IDCSj.S)),xor(Ri,H(BIOi.PWi))),T1')),IDi)
/\ K' := H(IDSi.xor(xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)),
xor(Ri,H(BIOi.PWi))).T1')
/\ M1' := xor(H(xor(xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)),
xor(Ri,H(BIOi.PWi))).IDSi.T1'), N1')
/\ Yi' := H(IDi.N1'.K'.T1')
/\ SEND(Pi'.M1'.Yi'.T1')
% Ui has freshly generated the value N1' for CSj
/\ witness(Ui, CSj, user_server_n1, N1')
% Verification phase
% Ui receives < M2, M3, T2 > from CSj via a public channel
3. State = 4 /\ RECV( xor(N2',H(IDSi.
xor(xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)),
xor(Ri,H(BIOi.PWi))).T1')), H(IDi.N1.SKij'.N2'.T2'), T2')
=|>
% Ui sends < M4, T3 > to CSj via a public channel
State' := 6 /\ T3' := new()
/\ M4' := xor(xor(SKij,H(IDSi.xor(xor(xor(Ri,H(BIOi.PWi)),
H(IDCSj.S)), xor(Ri,H(BIOi.PWi))).T1')),T3')
/\ SEND(M4'.T3')
% Ui's acceptance of the value N2 generated for Ui by CSj
/\ request(CSj, Ui, server_user_n2, N2')
end role
```

FIGURE 6. Role of a user U_i.

1) PRIVILEGED-INSIDER ATTACK [25]

U_i submits $\{ID_i, NPW_i\}$ to RC during the registration phase, where $NPW_i = H(BIO_i \| PW_i) \oplus r_i$. So, the privileged insider at RC does not come to know the password/biometric key of the registered user, $\mathcal{A}$ cannot obtain PW_i/BIO_i from NPW_i with the known fact that the collision resistant biohash one-way property, and $\mathcal{A}$ cannot guess PW_i/BIO_i from NPW_i with the absence of the knowledge on random strings r_i. Therefore, a privileged insider attack is restricted successfully in our proposed scheme.

2) PRESERVE USER ANONYMITY

Making use of the predefined threat model, we speculate the communicated messages between the parties such as U_i and CS_j is captured by $\mathcal{A}$ respectively. The messages $MSG_1 = \{P_i, M_1, Y_i, T_1\}$, $MSG_2 = \{M_2, M_3, T_2\}$ and $MSG_3 = \{M_4, T_3\}$ comprises of the valuable information such as the user's identity, under the computed values using

```
%%%%%%%%%%% Role for RC %%%%%%%%%%%%%%%%%%%%%
role rc (Ui, RC, CSj: agent,
% symmetric key between Ui and RC
        SKuirc : symmetric_key,
% H is hash function
         H : hash_func,
         BI : hash_func,
         SEND, RECV: channel(dy))
played_by RC
def=
 local State : nat,
     IDi, IDCSj, PWi, Ri, S : text,
     NPWi, BIOi, Ni, Ai, Cj, Zi, Li, Bi : text,
     IDSi, Pi, K, M1, Yi, N1, N2, M2, M3, M4 : text,
     SKij, T1, T2, T3, Tr : text

 const user_server_n1, server_user_n2,
       sub1, sub2, sub3 : protocol_id
init State := 0
 transition
% User registration phase
% RC receives <IDi, Ai> from Ui via a secure channel

1. State = 0 /\ RECV({IDi. xor(Ri,H(BIOi.PWi))}_SKuirc) =|>
State' := 1  /\ secret({PWi,S}, sub1, Ui)
             /\ secret({IDi,IDCSj}, sub2, {Ui,RC,CSj})
% RC sends < smart card > to Ui via a secure channel
           /\ Ni' := xor(H(IDi.S),xor(Ri,H(BIOi.PWi)))
           /\ Ai' := xor(xor(Ri,H(BIOi.PWi)),Cj)
           /\ SEND({Ai'.Ni'.H.BI}_SKuirc)
           /\ secret({S}, sub3, RC)
end role
```

FIGURE 7. Role of *RC*.

hash functions $Y_i = h(ID_i\|n_1\|K\|T_1)$ and XORed as $P_i = h((A_i \oplus NPW_i) \oplus T_1) \oplus ID_i$ operations. This shows the minimum knowledge of IDS_i, n_1 and biometric value BIO_i is required to breach this attack and derive the user's identity ID_i. Therefore, from this discussion we propose that our scheme successfully preserves user anonymity property.

3) PASSWORD/BIOMETRIC GUESSING ATTACK [27]

When it comes to the guessing of biometric key, it is computationally infeasible to derive the biometric key from the computed $NPW_i = h(PW_i\|b\|\alpha_i)$ value. Moreover, using the stored parameters of the smart card and the chances of breaking this verification $Z_i \stackrel{?}{=} h(ID_i\|A_i\|PW_i\|r_i)$ in Probabilistic Polynomial Time(PPT) is computationally infeasible task for an attacker. This impresses us that $\mathcal{A}$ fails to achieve any advantage regarding the guessing and computation of BIO_i and PW_i. Therefore, guessing of more than one secret in a single computation is computationally infeasible task. Hence, our scheme resists this attack.

4) USER IMPERSONATE ATTACK

This attack takes place if the attacker captures the transmitted messages such as $MSG_1 = \{P_i, M_1, Y_i, T_1\}, MSG_2 = \{M_2, M_3, T_2\}$ and $MSG_3 = \{M_4, T_3\}$. Now, to frame this attack in the real-time $\mathcal{A}$ needs to manipulate MSG_1. This can only happen if $\mathcal{A}$ able to compute $P_i = h((A_i \oplus NPW_i) \oplus T_1) \oplus ID_i$, $M_1 = h((A_i \oplus NPW_i)\|IDS_i\|T_1) \oplus n_1$, and $Y_i = h(ID_i\|n_1\|K\|T_1)$ with his credentials or any other parameters to make believe the cloud server CS_j as authentic. As $\mathcal{A}$ cannot compute $< P_i, M_i, Y_i >$, without the knowledge of $<ID_i, BIO_i, PW_i, r_i, n_1>$ and secret parameters $<s>$, this doesn't give any advantage to an attacker to impersonate

```
%%%%%%%%%%% Role for Cloud server CSj %%%%%%%%%%%%%%%%%%%%%
role server (Ui, RC, CSj : agent,
% symmetric key between Ui and RC
SKuirc : symmetric_key,
% H is hash function
H : hash_func,
BI : hash_func,
SEND, RECV: channel(dy))
played_by CSj
def=
local State : nat,
IDi, IDCSj, PWi, Ri, S : text,
NPWi, BIOi, Ni, Ai, Cj, Zi, Li, Bi : text,
IDSi, Pi, K, M1, Yi, N1, N2, M2, M3, M4 : text,
SKij, T1, T2, T3, Tr : text
const user_server_n1, server_user_n2,
sub1, sub2, sub3 : protocol_id
init State := 0
transition
% Logic phase
% CSj receives < M2, M3, T2 > from Ui via a public channel
1. State = 0 /\ RECV(xor(H(xor(xor(xor(xor(Ri,H(BIOi.PWi)),
H(IDCSj.S)),xor(Ri,H(BIOi.PWi))),T1')),IDi).
xor(H(xor(xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)),
xor(Ri,H(BIOi.PWi))).IDSi.T1'), N1').H(IDi.N1'.
H(IDSi.xor(xor(xor(Ri,H(BIOi.PWi)),H(IDCSj.S)),
xor(Ri,H(BIOi.PWi))).T1').T1') ) =|>
State' := 3 /\ secret({PWi,BIOi,Ai,K}, sub1, Ui)
/\ secret({IDi,IDCSj}, sub2, {Ui,RC,CSj})
/\ secret({S}, sub3, RC)
% Verification phase
/\ N2' := new() /\ T2' := new() /\ T3' := new()
/\ M2':=xor(N2',H(IDSi.xor(xor(xor(Ri,H(BIOi.PWi)),
H(IDCSj.S)), xor(Ri,H(BIOi.PWi))).T1'))
/\ SKij' := H(N1'.H(IDi.S).N2'.H(IDCSj.S).T2')
/\ M3' := H(IDi.N1'.SKij'.N2'.T2')
% CSj sends < M2, M3, T2 > to Ui via a public channel
/\ SEND(M2'. M3'. T2')
% CSj has freshly generated the value N2' for Ui
/\ witness(CSj, Ui, server_user_n2, N2')
% CSj receives < M4,T3 > from Ui via a public channel
2. State = 3
/\ RECV(xor(xor(SKij,H(IDSi.xor(xor(xor(Ri,H(BIOi.PWi)),
H(IDCSj.S)),xor(Ri,H(BIOi.PWi))).T1')),T3').T3') =|>
% CSj's acceptance of the value N2 generated for CSj by Ui
State' := 5 /\ N1' := new()
/\ request(Ui, CSj, user_server_n1, N1')
end role
```

FIGURE 8. Role of cloud servers CS_j.

the user. Therefore, the proposed scheme dissent the user impersonation attack.

5) RESIST THE SERVER CAPTURE ATTACK

As the servers are deployed on the hostile network, it is practical to vision that an attacker can physically capture the cloud server. Though the CS_j is captured by $\mathcal{A}$, the parameters stored on the server CS_j are different to the other servers when it comes to possess identity ID_{CS_j} and secret number s. This shows the masked secret stored on a particular server cannot be the same to any of the deployed servers. Thus even if some server is captured by $\mathcal{A}$ doesn't give any advantage to frame the attack. Hence $\mathcal{A}$ cannot pretend to be other cloud servers.

6) DENIAL-OF-SERVICE (DoS) ATTACKS [21]

It is practical in many ways where the user's smart card can be captured and extract the valuable information by applying the power analysis method. But to frame this attack $\mathcal{A}$ need to modify the password/biometric key of the user. If $\mathcal{A}$ is successful in updating the password/biometric without the notice of the user can only frame this DoS attack. But, in the update/change phase we observed that to make changes in

```
role session(Ui, RC, CSj : agent,
% symmetric key between Ui and RC
SKuirc : symmetric_key,
% H is hash function
H : hash_func,
BI : hash_func)
def=
local SN1, SN2, SN3, RV1, RV2, RV3 : channel (dy)
composition
user(Ui, RC, CSj, SKuirc, H, BI, SN1, RV1)
/\ rc(Ui, RC, CSj, SKuirc, H, BI, SN2, RV2)
/\ server(Ui, RC, CSj, SKuirc, H, BI, SN3, RV3)
end role
role environment()
def=
const ui, rc, csj: agent,
skuirc : symmetric_key,
h : hash_func,
bi : hash_func,
user_server_n1, server_user_n2,
sub1, sub2, sub3 : protocol_id
intruder_knowledge = {ui, rc, csj, h, bi}
composition
session(ui, rc, csj, skuirc, h, bi)
/\ session(i, rc, csj, skuirc, h, bi)
/\ session(ui, i, csj, skuirc, h, bi)
/\ session(ui, rc, i, skuirc, h, bi)
end role
goal
secrecy_of sub1
secrecy_of sub2
secrecy_of sub3
authentication_on user_server_n1
authentication_on server_user_n2
end goal
environment()
```

FIGURE 9. Role of session, goal and environment.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results
            /authentication.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 29.96s
  visitedNodes: 604 nodes
  depth: 6 plies
```

(a)

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results
            /authentication.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 6 states
  Reachable  : 0 states
  Translation: 0.09 seconds
  Computation: 0.00 seconds
```

(b)

FIGURE 10. The result of (a) OFMC backend (b) CI-AtSe backend.

the registered user's smart card the legitimate credentials such as user identity, password, biometric key are required. From the earlier discussion we already seen that guessing of password/biometric is computationally infeasible for an attacker. Hence, $\mathcal{A}$ cannot frame this DoS attack.

7) ENSURES MUTUAL AUTHENTICATION [27], [28]

This attack can be defended into two cases:

1. The authenticity of the user is validated, CS_j receives the messages and then checks the validity of Y_i to authenticate U_i. If this verification is successful, CS_j believes U_i.
2. The authenticity of the cloud server is validated, U_i receives the message and then checks the validity of M_3 to authenticate CS_j. If this verification is successful, U_i believes CS_j.

This validation gives an clear insight about the ensuring mutual authentication.

8) RESIST CLOUD SERVER COMPROMISE ATTACK

In proposed scheme, we assume if the cloud server CS_j is compromised than one may feel that all the past session keys as well as future keys will be compromised. But, this is not true with our scheme, as attacker can only compromise a cloud server CS_j but it is not easily to retrieve the session key. Before computing or retrieving the session key $SK_{ij} = h(n_1\|CU_i\|n_2\|C_j\|T_2)$, the attacker has to get pass the verification $h(s\|ID_i)$ in Rec_U, once this verification holds, then only attacker will be allowed to perform computations such as $CU_i = h(ID_i\|s)$, $K = h(CU_i\|C_j\|T_1)$, $n_1 = M_1 \oplus h(C_j\|CU_i\|T_1)$ then, verify $Y_i \stackrel{?}{=} h(ID_i\|n_1\|K\|T_1)$. But, though the cloud server is compromised attacker cannot know the long-term secret key s without which attacker cannot compute the above computations and also cannot retrieve the session key and compromise the future sessions. Thus the proposed scheme is secure to perfect secrecy attack. i.e., Attacker cannot compromise the long term secrets. Therefore, the future communications and the session keys are secure.

9) SESSION KEY AGREEMENT [27], [28]

As discussed above, once the participants authenticates each other, this is due to the acknowledgment of the participants and establish a valid session key $SK_{ij} = h(n_1\|IDS_i\|n_2\|C_j\|T_2)$. Thus, our proposed scheme agree on a session key.

10) RESISTANT TO SESSION KEY DISCLOSURE ATTACK

Assume that an adversary $\mathcal{A}$ listens to the ongoing communication and records login message $MSG_1 = \{P_i, M_1, Y_i, T_1\}$ of U_i, authentication request $MSG_2 = \{M_2, M_3, T_2\}$ and $MSG_3 = \{M_4, T_3\}$. Firstly, $\mathcal{A}$ cannot extract $SK_{ij} = h(n_1\|CU_i\|n_2\|C_j\|T_2)$ from M_3 a M_3 is secured by the collision-resistant one-way hash function $h(\cdot)$. Secondly, to successfully generate SK_{ij}, $\mathcal{A}$ must have the knowledge of the unique session-specific temporary secret $h(ID_i\|s)$ which can only be computed by the cloud server CS_j. The parameters ID_i and s the secret of CS_j are never revealed or transmitted and cannot be extracted by adversary as they are encrypted by one-way hash function $h(\cdot)$.

11) PERFECT FORWARD SECRECY

Assume that an adversary $\mathcal{A}$ listens to the ongoing communication and records login message $MSG_1 = \{P_i, M_1, Y_i, T_1\}$ of U_i, authentication request $MSG_2 = \{M_2, M_3, T_2\}$ and $MSG_3 = \{M_4, T_3\}$. Firstly, it is clear from our above discussion that $\mathcal{A}$ cannot extract $SK_{ij} = h(n_1\|CU_i\|n_2\|C_j\|T_2)$ from the transmitted messages. Secondly, to successfully generate SK_{ij}, $\mathcal{A}$ must have the knowledge of the unique

TABLE 4. Computation cost analysis [40].

Notation	Description	≈ execution time
T_h	One-way hash function	$0.0023ms$
T_Ω	Symmetric key encryption/decryption	$0.0046ms$
T_{EPM}	Elliptic curve point multiplication	$2.2260ms$
T_{BIO-H}	Biohashing key	$2.2260ms$
T_{FE}	Biometric Fuzzy extractor	$2.2260ms$
T_{EPA}	Time for performing an elliptic curve point addition	$0.0288ms$
T_{EXP}	1024-bit modular exponentiation	$3.8500ms$
T_{EINV}	Time of performing an 160-bit modular inversion	$0.005565ms$

TABLE 5. Comparison of security features.

Security attributes ↓ / Schemes →	[3]	[29]	[36]	[46]	[37]	Our scheme
Online/Offline						
Password guessing attack	√	√	×	√	√	√
Privileged-insider attack	×	√	×	√	√	√
User anonymity preservation	√	×	√	√	√	√
Traceability preservation	×	√	×	√	√	√
Detection for unauthorized login	√	√	√	√	√	√
Stolen mobile/smart card attack	√	√	√	√	√	√
Suitable for IoT environments	×	×	×	√	×	√
Denial-of-service attack	√	×	√	√	√	√
Mutual authentication	√	√	√	√	√	√
Man-in-the-middle attack	√	√	√	√	√	√
Forward Secrecy	×	√	×	√	×	√
Explicit Key Authentication	√	×	√	√	√	√
ESL attack	×	×	√	×	×	√
Replay attack	×	√	√	√	√	√
Impersonation attacks	×	×	√	√	×	√
Server masquerade attack	√	×	√	×	√	√
Revocability	√	√	×	√	√	√
Freely password/biometric change	√	×	√	√	√	√

session-specific temporary secret $h(ID_i\|s)$ which can only be computed by the cloud server CS_j. The parameters ID_i and s the secret of CS_j are never revealed or transmitted and cannot be extracted by adversary as they are encrypted by one-way hash function $h(\cdot)$. This ensures the fact that $\mathcal{A}$ cannot achieve any of the session keys which were established in the past or going to establish in the future. This shows, $\mathcal{A}$ cannot compromise the session key. Hence, the scheme is secure to preserve the perfect forward secrecy attack.

VII. PERFORMANCE COMPARISON WITH RELATED SCHEMES

This section deals with the description of the performance of the proposed scheme which is compared among the other relative schemes in terms of security/functionality features, smart card storage cost, communication costs, and computational costs. This performance comparisons is essential to give the best understandability of the proposed scheme. This evaluation gives an insight into the effectiveness of the proposed scheme.

A. SECURITY FEATURES COMPARISON

In Table 5, deals with the comparison of the security and functionality features which represents that our proposed scheme restricts the attacks from the attacker with much effectiveness and provides the complete security of the system in comparison to other related schemes [3], [29], [36], [37], [46]. It is worthy noted that in comparison to the existing schemes,

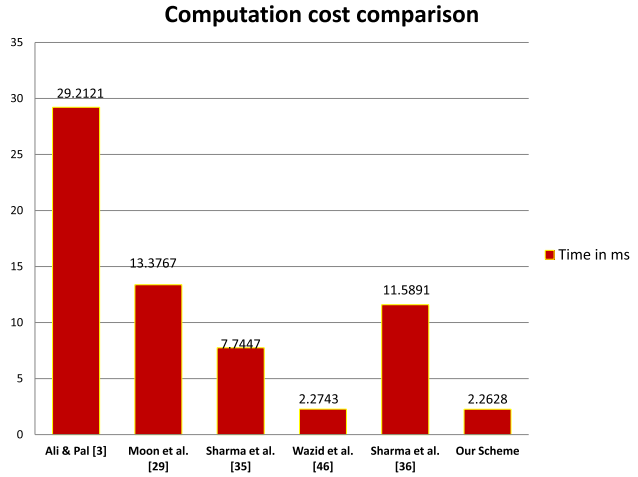

FIGURE 11. Comparison of computation cost.

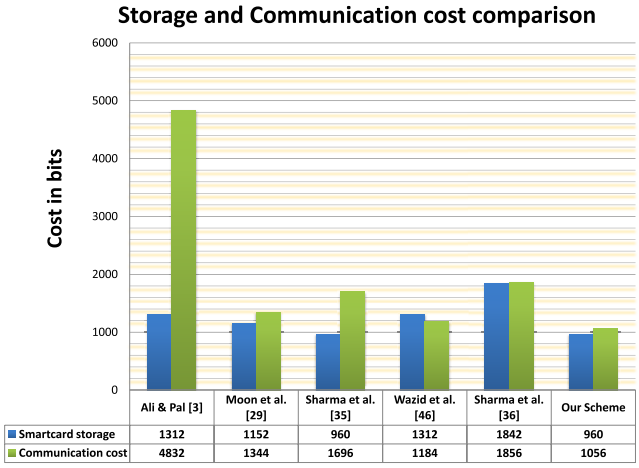

FIGURE 12. Storage and Communication cost comparison.

our scheme resists various known attacks and also efficiently preserves various security features.

B. PERFORMANCE ANALYSIS

In Table 4, the time consumed by the cryptographic one-way hash function $T_h = 0.0023ms$ and the symmetric encryption/decryption operations $T_\Omega = 0.0046ms$ were considered. In Table. 6, the time consumption cost while the login and authentication phases and also in Table. 7 the communication sizes between our scheme and other schemes have been compared. We assume that the output of the one-way hash function $h(\cdot)$ is 128 bits, if we use SHA-1 hashing algorithm [1] and symmetric encryption as 256 bits. Further, we assume that each timestamp, random nonce/random number, identity of (U_i, RC, CS_j)'s is 160 bits in length.

We analyze the detailed results to give a better insight of the performance analysis, as follows:

TABLE 6. The performance comparison among our scheme and other schemes.

Scheme		Ali & Pal [3]	Moon *et al.* [29]	Sharma *et al.* [36]	Wazid *et al.* [46]	Sharma *et al.* [37]	Our
Computation cost and time for Login and Authentication	U_i :	$1T_{BIO-H} + 5T_h +$ $7T_{EPM} + 3T_{EPA}$	$5T_h + 2T_{EPM}$ $1T_{FE}$	$11T_h + 2T_{EXP}$	$12T_h + 1T_{FE}$	$11T_h + 1T_{EXP}$	$9T_h + 1T_{BIO-H}$
	RA :	$3T_\Omega + 4T_h +$ $2T_{EPM} + 3T_{EPA}$	– –	– –	– –	– –	– –
	$S/S_j/CS_j$:	$T_\Omega + 2T_h +$ $3T_{EPM} + 2T_{EPA}$	$4T_h + 2T_{EPM}$ $+1T_{FE}$	$6T_h + 1T_{EINV}$	$9T_h$	$6T_h + 2T_{EXP}$	$7T_h$
	$Total$ (ms) :	$1T_{BIO-H} + 4T_\Omega + 11T_h +$ $12T_{EPM} + 8T_{EPA}$ ≈ 29.2121	$9T_h + 4T_{EPM}$ $+2T_{FE}$ ≈ 13.3767	$17T_h + 2T_{EXP}$ $+1T_{EINV}$ ≈ 7.7447	$21T_h + 1T_{FE}$ ≈ 2.2743	$17T_h + 3T_{EXP}$ ≈ 11.5891	$16T_h + 1T_{BIO-H}$ ≈ 2.2628

TABLE 7. Comparison of smart card storage cost and communication cost.

Scheme	Smartcard Storage cost	Messages during AKA	Communication cost
Ali & Pal [3]	1312 bits	3	4832 bits
Moon *et al.* [29]	1152 bits	2	1344 bits
Sharma *et al.* [36]	960 bits	2	1696 bits
Wazid *et al.* [46]	1312 bits	3	1184 bits
Sharma *et al.* [37]	1824 bits	3	1856 bits
Our scheme	960 bits	3	1056 bits

- In comparison to earlier proposed schemes, the proposed scheme consumes much lesser computation cost, such as ≈2.2628ms as discussed in Table 6. It is very clear from the comparison our scheme results more efficient than [3], [29], [36], [37], [46] with computation cost ≈29.2121, ≈13.3767, ≈7.7447, ≈2.2743, ≈11.5891 ms. Furthermore, the compared scheme proves to be vulnerable to achieve security requirements as shown in the Table. 5. A graphical representation of the comparison is given in Figure.11. Thus, our scheme proves to be more reliable due to the security and performance.
- From Table.7, we were able to illuminate the communication cost consumption of our scheme takes much less than the existing schemes [3], [29], [36], [37], [46].
- As shown in Table.7, we were able to show our scheme and Sharma *et al.* scheme takes much less storage space than [3], [29], [37], [46] schemes. A graphical representation of the comparison is given in Figure.12 which gives a greater insight.
- The most considerable and acceptable part is the security. Our scheme draws out the security features and presents the drawbacks of the other compared schemes. Furthermore, our scheme is proven secure based on ROR model, BAN Logic and AVISPA which indeed boost the security strength of our argument.

Therefore, Tables 5, 6 and 7, justifies that our scheme is the best among the other five schemes in terms of security, computation cost and communication cost.

VIII. CONCLUSION

In recent years, many applications stores the valuable data on the servers of the government, which were developed of ICT based E-governance. The citizens are allowed to access the data in order to smooth run of the E-governance by availing the applications provided by the government of smartcity. In such situations, user authentication in accessing the data (applications) provided by the government becomes a crucial issue as to avoid unattended access. To address this issue, we have proposed a scheme to facilitate a user whoever wishes to access the data provided by government has to register to the server of the government and avail the credentials required to access. The formal analysis on our proposed scheme is proved based on ROR model. To strengthen our proposed scheme, we have provided informal analysis to ensure the security and functional attributes. Using AVISPA, we conduct the formal security verification and the result proves to be resistant to several active and passive attacks. Furthermore, our scheme proves to be efficient in comparison to the compared schemes. Based on the computation cost comparison, communication and storage bits comparison with the aforementioned schemes.

REFERENCES

[1] *Secure Hash Standard, FIPS PUB 180-1, National Institute of Standards and Technology (NIST)*, U.S. Dept. Commerce, Washington, DC, USA, Apr. 1995. Accessed: Jul. 2015. [Online]. Available: http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf
[2] M. Abdalla, P.-A. Fouque and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Public Key Cryptography—PKC* (Lecture Notes in Computer Science), vol. 3386. Berlin, Germany: Springer, 2005, pp. 65–84.
[3] R. Ali and A. K. Pal, "An efficient three factor–based authentication scheme in multiserver environment using ECC," *Int. J. Commun. Syst.*, vol. 31, no. 4, p. e3484, 2018.
[4] A.-V. Anttiroiko, P. Valkama, and S. J. Bailey, "Smart cities in the new service economy: Building platforms for smart services," *AI Soc.*, vol. 29, no. 3, pp. 323–334, 2014.
[5] AVISPA. *Automated Validation of Internet Security Protocols and Applications*. Accessed: Jan. 2015. [Online]. Available: http://www.avispa-project.org/
[6] AVISPA. *SPAN—The Security Protocol Animator for AVISPA*. Accessed: Jan. 2018. [Online]. Available: http://www.avispa-project.org
[7] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.
[8] A. Caragliu, C. Del Bo, and P. Nijkamp, "Smart cities in europe," *J. Urban Technol.*, vol. 18, no. 2, pp. 65–82, 2011.
[9] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2016.
[10] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "Secure biometric-based authentication scheme using chebyshev chaotic map for multi-server environment," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 824–839, Sep./Oct. 2016, doi: 10.1109/TDSC.2016.2616876.

[11] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 14, no. 6, pp. 10081–10106, 2014.
[12] Y. Choi, Y. Lee, and D. Won, "Security improvement on biometric based authentication scheme for wireless sensor networks using fuzzy extraction," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 1, p. 8572410, 2016.
[13] H. Chourabi *et al.*, "Understanding smart cities: An integrative framework," in *Proc. 45th Hawaii Int. Conf. Syst. Sci. (HICSS)*, Jan. 2012, pp. 2289–2297.
[14] A. Coe, G. Paquet, and J. Roy, "E-governance and smart communities: A social learning challenge," *Social Sci. Comput. Rev.*, vol. 19, no. 1, pp. 80–93, 2001.
[15] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, "A dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1646–1656, 2012.
[16] 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.
[17] C. Harrison *et al.*, "Foundations for smarter cities," *IBM J. Res. Develop.*, vol. 54, no. 4, pp. 1–16, Jul./Aug. 2010.
[18] A. Hoda, A. Roy, and S. Karforma, "Application of ecdsa for security of transaction in e-governance," in *Proc. 2nd Nat. Conf. Comput. Syst. (NaCCS)*, 2012, pp. 281–286.
[19] S. Jangirala, S. Mukhopadhyay, and A. K. Das, "A multi-server environment with secure and efficient remote user authentication scheme based on dynamic ID using smart cards," *Wireless Pers. Commun.*, vol. 95, pp. 2735–2767, Apr. 2017.
[20] A. T. B. Jin, D. N. C. Ling, and A. Goh, "BioHashing: Two factor authentication featuring fingerprint data and tokenised random number," *Pattern Recognit.*, vol. 37, no. 11, pp. 2245–2255, Apr. 2004.
[21] J. Jung, J. Kim, Y. Choi, and D. Won, "An anonymous user authentication and key agreement scheme based on a symmetric cryptosystem in wireless sensor networks," *Sensors*, vol. 16, no. 8, p. E1299, 2016.
[22] S. Kalra and S. K. Sood, "Secure authentication scheme for IoT and cloud servers," *Pervasive Mobile Comput.*, vol. 24, pp. 210–223, Dec. 2015.
[23] S. Kim, H. J. Kim, and H. Lee, "An institutional analysis of an E-government system for anti-corruption: The case of OPEN," *Government Inf. Quart.*, vol. 26, no. 1, pp. 42–50, 2009.
[24] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 1666, M. Wiener, Ed. Berlin, Germany: Springer-Verlag, 1999, pp. 388–397.
[25] S. Kumari, X. Li, F. Wu, A. K. Das, H. Arshad, and M. K. Khan, "A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps," *Future Gener. Comput. Syst.*, vol. 63, pp. 56–75, Oct. 2016.
[26] C. T. Li, C. Y. Weng, and C. C. Lee, "An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks," *Sensors*, vol. 13, pp. 9589–9603, Jul. 2013.
[27] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K.-K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things environments," *J. Netw. Comput. Appl.*, vol. 103, pp. 194–204, Feb. 2018.
[28] X. Li, J. Niu, J. Liao, and W. Liang, "Cryptanalysis of a dynamic identity-based remote user authentication scheme with verifiable password update," *Int. J. Commun. Syst.*, vol. 28, pp. 374–382, Jan. 2015.
[29] J. Moon, D. Lee, J. Jung, and D. Won, "Improvement of efficient and secure smart card based password authentication scheme," *Int. J. Netw. Secur.*, vol. 19, no. 6, pp. 1053–1061, 2017.
[30] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. Roy. Soc. London. A. Math. Phys. Sci.*, vol. 426, no. 1871, pp. 233–271, 1989.
[31] A. Roy and S. Karforma, "Risk and remedies of e-governance systems," *Oriental J. Comput. Sci. Technol.*, vol. 4, no. 2, pp. 329–339, 2011.
[32] A. Roy and S. Karforma, "A survey on E-governance security," *Int. J. Comput. Eng. Comput. Appl.*, vol. 8, no. 1, pp. 50–62, 2011.
[33] A. Roy and S. Karforma, "UML based modeling of ECDSA for secured and smart E-governance system," in *Proc. Comput. Sci. Inf. Technol. (CS IT-CSCP), Nat. Conf. Adv. Comput. Eng. Res. (ACER) Organized Global Inst. Manage. Technol.*, 2013, pp. 207–222.
[34] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "On the design of provably secure lightweight remote user authentication scheme for mobile cloud computing services," *IEEE Access*, vol. 5, no. 1, pp. 25808–25825, 2017.
[35] G. Sharma and S. Kalra, "A novel scheme for data security in cloud computing using quantum cryptography," in *Proc. Int. Conf. Adv. Inf. Commun. Technol. Comput.*, 2016, p. 37.
[36] G. Sharma and S. Kalra, "A secure remote user authentication scheme for smart cities E-governance applications," *J. Reliable Intell. Environ.*, vol. 3, no. 3, pp. 177–188, 2017.
[37] G. Sharma and S. Kalra, "Advanced multi-factor user authentication scheme for E-governance applications in smart cities," *Int. J. Comput. Appl.*, pp. 1–16, Mar. 2018.
[38] G. Sharma and S. Kalra, "Identity based secure authentication scheme based on quantum key distribution for cloud computing," *Peer-Peer Netw. Appl.*, vol. 11, no. 2, pp. 220–234, 2018.
[39] S. Shunmuganathan, R. D. Saravanan, and Y. Palanichamy, "Secure and efficient smart-card-based remote user authentication scheme for multi-server environment," *Can. J. Elect. Comput. Eng.*, vol. 38, no. 1, pp. 20–30, 2015.
[40] J. Srinivas, A. K. Das, N. Kumar, and J. Rodrigues, "Cloud centric authentication for wearable healthcare monitoring system," *IEEE Trans. Dependable Secure Comput.*, to be published, doi: 10.1109/TDSC.2018.2828306.
[41] J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, "Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial Internet of Things," *IEEE Trans. Dependable Secure Comput.*, to be published.
[42] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," *Ad Hoc Netw.*, vol. 54, pp. 147–169, Jan. 2017.
[43] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "A self-verifiable password based authentication scheme for multi-server architecture using smart card," *Wireless Pers. Commun.*, vol. 96, no. 4, pp. 6273–6297, Oct. 2017.
[44] D. von Oheimb, "The high-level protocol specification language HLPSL developed in the EU project AVISPA," in *Proc. APPSEM Workshop*, 2005, pp. 1–17.
[45] D. Wang, D. He, P. Wang, and C.-H. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 4, pp. 428–442, Jul./Aug. 2015.
[46] M. Wazid, A. K. Das, S. Kumari, X. Li, and F. Wu, "Provably secure biometric-based user authentication and key agreement scheme in cloud computing," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4103–4119, 2016.

SAUD S. ALOTAIBI received the bachelor's degree in computer science from King Abdul Aziz University, Jeddah, Saudi Arabia, in 2000, the master's degree in computer science from King Fahd University, Dhahran, Saudi Arabia, in 2008, and the Ph.D. degree in computer science from Colorado State University, Fort Collins, CO, USA, in 2015. From 2009 to 2010, he was the Deputy of the IT Center for E-Government and Application Services, Umm Al-Qura University, Makkah, Saudi Arabia.

He is currently an Assistant Professor with the Department of Information Systems, College of Computer and Information Systems, working with the Deanship of Information Technology to improve the IT services that are provided to the Umm Al-Qura University. His current research interests include emotional intelligence, data mining, natural language processing, machine learning, deep learning, computer networks, wireless sensor networks, and network security.