Engineering Reports

WILEY

Engineering Reports Open Access

RESEARCH ARTICLE OPEN ACCESS

Anonymous Authentication Scheme Based on Physically Unclonable Function and Biometrics for Smart Cities

Vincent Omollo Nyangaresi[1,2] | Ahmad A. AlRababah[3] | Ganesh Keshaorao Yenurkar[4] | Ravikumar Chinthaginjala[5] | Muhammad Yasir[6]

[1]Department of Computer Science & Software Engineering, Jaramogi Oginga Odinga University of Science & Technology, Bondo, Kenya | [2]Department of Applied Electronics, Saveetha School of Engineering, SIMATS, Chennai, Tamilnadu, India | [3]Faculty of Computing and Information Technology in Rabigh, King Abdulaziz University, Rabigh, Saudi Arabia | [4]Yeshwantrao Chavan College of Engineering, Nagpur, India | [5]School of Electronics Engineering, Vellore Institute of Technology, Vellore, India | [6]College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao, China

Correspondence: Vincent Omollo Nyangaresi (vnyangaresi@jooust.ac.ke)

Received: 7 May 2024 | **Revised:** 13 November 2024 | **Accepted:** 25 November 2024

Funding: The authors received no specific funding for this work.

Keywords: anonymity | authentication | biometrics | privacy | PUF | security | sensors | smart city

ABSTRACT

Smart cities amalgamate technologies such as Internet of Things, big data analytics, and cloud computing to collect and analyze large volumes of data from varied sources which facilitate intelligent surveillance, enhanced energy management systems, and environmental monitoring. The ultimate goal of these smart cities is to offer city residents with better services, opportunities, and quality of life. However, the vulnerabilities in the underlying smart city technologies, interconnection of heterogeneous devices, and transfer of data over the open public channels expose these networks to a myriad of security and privacy threats. Therefore, many security solutions have been presented in the literature. However, the majority of these techniques still have numerous performance, privacy, and security challenges that need to be addressed. To this end, we present an anonymous authentication scheme for the smart cities based on physically unclonable function and user biometrics. Its formal security analysis using the Real-Or-Random (ROR) model demonstrates the robustness of the negotiated session key against active and passive attacks. In addition, the informal security analysis shows that it supports salient functional and security features such as mutual authentication, key agreement, perfect key secrecy, anonymity, and untraceability. It is also shown to withstand typical smart city threats such as side-channeling, offline guessing, session key disclosure, eavesdropping, session hijacking, privileged insider, and impersonation attacks. Moreover, comparative performance shows that it incurs the lowest energy and computation costs at relatively low communication overheads.

1 | Introduction

A smart city is a metropolitan region that utilizes data-driven technologies to streamline urban services and enhance sustainability, efficiency as well as citizens' quality of life. Such technologies include artificial intelligence, Internet of Things (IoT), cloud computing, and big data analytics [1]. With the help of these technologies, smart cities collect, process and analyze high volumes of data emanating from a wide range of sources such as social media, mobile devices, and sensors. This enables smart cities to enhance their services and operations in terms of reduced crime rates, energy efficiency, improved citizen engagement and traffic flow. It also facilitates governance processes and city management digitization [2] with reduced negative impacts on the

environments [3]. For instance, the IoT deployed in smart cities helps create smart and people-centric industries and cities which facilitate the development of sustainable and smart environments [4]. This is reflected in the resulting automated transportation, water distribution, intelligent surveillance, enhanced energy management systems, urban security, and environmental monitoring [4–6]. Ultimately, this results in cities that are responsive, sustainable, and more connected to offer informed decisions, manage assets and resources efficiently, and hence provide city residents with better services, opportunities, and quality of life [7, 8].

Owing to the faster growth of many cities, advanced technologies need to be adopted to boost citizen safety, lifestyle, and health. This is accomplished by addressing issues such as traffic management, pollution, digital security, energy efficiency, waste management, and street lights management. In light of this, smart cities interconnect millions of sensors and devices which generate and share massive data [9–11]. This enables the automation of governance, infrastructure, policies, and services such as agriculture, healthcare, education, transportation [12], industrialization, and constructions. Although smart cities offer a wide range of services in an efficient and convenient manner, they are vulnerable to numerous privacy and security threats. For instance, some of the collected data include sensitive user records which must be protected from illegal access and transfer. In addition, the data collected from multiple sources is utilized to make critical decisions regarding the management of the smart city ecosystem. As such, the veracity, authenticity, and integrity of this data must be upheld to prevent any unauthorized access and usage [13]. However, the usage of public channels for data exchange among the sensors and other smart city devices exposes the collected data to many threats [14, 15]. As such, reliability during data collection and transfer by IoT is crucial [16].

Another serious challenge in IoT-based smart cities is the interconnection of numerous heterogeneous devices which increase the surface from which attacks can be initiated. Any successful attack on any of these devices can have serious ramification to the smart city and the safety of its residents. In addition, various communication technologies are deployed in the management of data, networking systems as well as cloud computing [17, 18]. Although many schemes have been developed for smart cities, security protocols interoperability among diverse technologies presents some challenges. Since the majority of the sensor devices are limited in terms of battery and computation power, highly complex security solutions are unsuitable. There is therefore need for the development of lightweight security and privacy-preserving schemes for the smart cities. There is also a need for the detection of runtime anomalies in the IoT environment so as to thwart adversarial attacks. Unfortunately, most of the anomaly detection techniques are inadequate when deployed in this dynamic environment [19]. As such, there is need for innovative techniques to handle this complexity.

1.1 | Motivation

Smart cities have been developed to offer crucial services such as resource utilization, healthcare monitoring [20], and resource management systems. This requires that vast amounts of data be collected from various sources such as cameras, sensors, and other smart devices [21]. Some of the collected data items are sensitive and personal in nature since they are related to citizen habits, location, and behavior. In spite of this sensitivity, open public channels are deployed to exchange data in smart cities. In addition, some of the devices and technologies deployed in smart cities may have vulnerabilities that can be exploited by the attackers to gain unauthorized access, manipulate the systems or steal data. Moreover, the numerous and heterogeneous sensors and devices interconnected in smart cities increase the surface from which attacks can be launched. Any successful attack can have serious economic and social ramifications to smart city residents. For instance, malicious nodes in military, agriculture, and healthcare surveillance systems can mishandle network data, leading to privacy violations. All these issues point to the need for robust security and privacy-preserving techniques that can ensure authentication, data integrity and uphold trust among smart city residents [22, 23]. Since the majority of the sensors and devices in smart cities are limited in terms of transmission range, energy, and battery life, these security solutions need to be highly lightweight to boost efficiency.

1.2 | Contributions

The main contributions acclaimed in this study include the following:

- We leverage user biometrics and physical unclonable function (PUF) to develop an authentication scheme that is shown to withstand physical and side-channeling attacks among other smart city threats.
- Time-stamping is deployed to all messages exchanged over public communication channels so as to preserve the freshness of these messages and avert replay attacks.
- Elaborate formal security analysis is carried out using the ROR model to show the robustness of the negotiated session key against active and active adversarial attacks.
- We execute extensive informal security analysis to demonstrate that our scheme withstands typical smart city attacks such as offline guessing, session key disclosure, eavesdropping, session hijacking, ESL, replays, forgery, MitM, privileged insider, physical, side-channeling, and impersonations. In addition, our scheme is shown to support mutual authentication, key agreement, perfect key secrecy, anonymity, and untraceability.
- We carry out a comparative performance evaluation to show that our protocol incurs the least computation overheads and consumes the least amount of energy. In addition, the communication overhead of our scheme is shown to be relatively lower.

1.3 | Paper Structure

The rest of this paper is organized as follows: Section 2 discusses the related works in this domain while Section 3 presents the proposed scheme. On the other hand, Section 4 presents the security analysis while Section 5 describes performance evaluation.

Toward the end of this paper, Section 6 presents the conclusion and future research scope.

2 | Related Works

Security and privacy issues in smart cities have attracted a lot of traction from academia and the industry. As such, a myriad of schemes have been developed for these interconnected smart cities [24]. For instance, an identity-based anonymous authentication scheme is presented in [25]. Although this scheme offers efficiency and better security, identity-based schemes are vulnerable to key escrow threats [26]. To address this challenge, blockchain-based security solutions have been introduced to offer decentralized security [27, 28]. For instance, a blockchain-based authentication scheme between the sensor node and the base stations is developed in [7] while a secure and privacy protocol leveraging on blockchain is introduced in [29]. Similarly, the protocol in [30] utilizes blockchain to manage trust and routing in wireless networks. However, blockchain technology comes with heavy computation, communication, and storage requirements [31]. Therefore, a lightweight authentication scheme is developed in [32]. However, this protocol is susceptible to physical capture attacks through which its stored secret tokens can be retrieved [1].

To preserve anonymity during the communication process, a privacy-preserving technique is presented in [7]. However, this scheme is susceptible to impersonation and password-guessing attacks. In addition, it fails to provide mutual authentication [7]. Similarly, the biometric-based protocols in [33, 34] fail to provide mutual authentication. In addition, the scheme in [33] is vulnerable to Denial of Service (DoS) while the protocol in [34] cannot offer anonymity and is susceptible to password-guessing attacks [7]. To offer mutual authentication, accountability, privacy, and protect against Sybil attacks, a pseudonym-based technique is developed in [35]. Unfortunately, this scheme cannot support pseudonym unlinkability since the pseudonyms can be linked to the same credential holder [2]. In addition, it is not scalable in smart city environments where users may need periodic pseudonym switching. Therefore, improved security schemes are introduced in [36–41]. However, the scheme in [36] has not been evaluated against attacks such as eavesdropping, ephemeral secret leakage, and side-channeling attacks. On its part, the protocol [37] is vulnerable to DoS and impersonation attacks [7]. In addition, it cannot offer mutual authentication and fails to preserve anonymity. Similarly, the technique in [38] cannot support mutual authentication and untraceability, and is susceptible to impersonation attacks [7]. Although the protocol in [39] offers mutual authentication, it cannot withstand sensor impersonation, Man-in-the-Middle (MitM), sensor capture and offline guessing attacks [36]. Similarly, the scheme in [41] is susceptible to offline guessing and Known Session-Specific Temporary Information (KSSTI) attacks. In addition, it fails to support backward key secrecy [42]. Regarding the technique in [40], its failure to offer mutual authentication and anonymity are its major weaknesses, together with its vulnerability to impersonation attacks [7].

To improve scalability and reduce both latency and overheads, a deep learning model is presented in [43]. However, this model has reduced trustworthiness and lacks social control [44]. On its part, the context-based trust model developed in [45] incurs heavy computation costs due to the need for the processing of huge volumes of contextual data. Similarly, the quantum-inspired approach in [46] has huge computation overheads due to the required quantum computing. The smart city security framework presented in [47] can prevent eavesdropping, replay, node capture, spoofing, and side-channel attacks. However, it exhibits high redundancies, key generation and network overheads [44]. Similarly, the Rivest-Shamir-Adleman (RSA)-based authentication scheme in [48] is computationally extensive during encryption and decryption. Therefore, lightweight authentication approaches are introduced in [49, 50]. However, these schemes have not been evaluated against attack vectors such as session hijacking and side-channeling. Table 1 gives a summary of the security, performance, and privacy gaps in the existing security techniques.

Based on the above discussion, it is clear that securing data exchanges in smart cities is a necessary but intricate task. Most of the current techniques for security and privacy protection are shown to have numerous performance and security challenges that need to be fixed. In addition, the current anomaly detection methods have been noted to be inadequate when faced with the smart city dynamism and complexity. Our scheme is demonstrated to address the majority of these security, performance, and privacy challenges.

3 | Proposed Scheme

In this section, we present the mathematical preliminaries as well as the major phases of our protocol.

3.1 | Mathematical Preliminaries

The physical unclonable function and fuzzy extraction are the main building blocks of the proposed scheme. As such, the following subsections describe their mathematical formulations.

3.1.1 | Physical Unclonable Function

The physical unclonable function (PUF) is designed to generate some output for any input data such as biometrics based on the intrinsic physical characteristics of the devices. Since PUF is produced through the generation of nano-scale variations in the manufacturing process of the Integrated Circuit (IC) chips, it is cumbersome to entirely clone PUFs. As such, PUFs can protect against attacks such as side-channeling, tampering and cloning while at the same time offering unpredictability, reliability and uniqueness. Suppose that C is the input challenge and P is the PUF function and R is the output response. Therefore,

$$R = P(C) \tag{1}$$

In other words, for any $j \in \mathbb{N}$, an n-bit challenge C_j produces a unique m-bit response R_j. Basically, any PUF instance expressed as follows.

$$f_{PUF} : C_j \rightarrow R_j \tag{2}$$

TABLE 1 | Summary of existing works.

Protocol	Technique	Gaps
Gupta et al. [7]	Blockchain	High storage and computation costs
Wu et al. [25]	Identity-based	Vulnerable to key escrow
El Bekkali, Essaaidi, and Boulmalf [29]	Blockchain	High storage and computation costs
Awan et al. [30]	Blockchain	High storage and computation costs
Nikooghadam et al. [32]	ECC	Prone to physical capture attacks
Ghahramani, Javidan, and Shojafar [33]	Biometrics	Fail to offer mutual authentication
		Vulnerable to DoS
Bera et al. [34]	Biometrics	Fail to offer mutual authentication
		Cannot provide anonymity
		Susceptible to password guessing
Maram et al. [35]	Identity-based	Fails to provide unlinkability
		Not scalable
Lee, Oh, and Park [36]	PUF	Not evaluated against eavesdropping, ephemeral secret leakage, and side-channeling
Vijayakumar et al. [37]	Bilinear pairing	Prone to DoS and impersonation attacks
		High computation complexity
		Cannot offer mutual authentication and anonymity
Xie et al. [38]	ECC	Fails to support mutual authentication and untraceability
		Susceptible to impersonation
Yuanbing, Wanrong, and Bin [39]	ECC	Vulnerable to sensor impersonation, MitM, sensor capture, and offline guessing
Xia et al. [40]	ECC	Cannot offer mutual authentication and anonymity
		Susceptible to impersonations
Lu et al. [41]	ECC	Does not support backward key secrecy
		Vulnerable to offline guessing and KSSTI attacks
Singh, Jeong, and Park [43]	Deep learning	Reduced trustworthiness
		Lacks social control
Altaf et al. [45]	Context-based	Heavy computation costs
Abd El-Latif et al. [46]	Quantum-inspired	Huge computation overheads
Wang et al. [47]	Physical layer security	High redundancies
		Large key generation and network overheads
Dharminder, Mishra, and Li [48]	RSA	High computation costs
Chaudhary et al. [49]	PUF	Evaluation against session hijacking and side-channeling is missing
Chaudhary et al. [50]	Hash function and bitwise XOR	Evaluation against session hijacking and side-channeling is missing

where $C_j \in \{0,1\}^n$ and $R_J \in \{0,1\}^m$. Under these conditions, the following PUF definitions hold.

Definition 1. Considering PUF instances f_{PUF}^{A} and f_{PUF}^{B}, input challenge C_A and C_B where $A, B \in \mathbb{N}$, the PUF uniqueness property is mathematically denoted as follows.

$$f_{PUF}^{A}(C_A) \neq f_{PUF}^{A}(C_B) \tag{3}$$

$$f_{PUF}^{A}(C_A) \neq f_{PUF}^{B}(C_A) \tag{4}$$

In other words, PUFs cannot produce the same response given different input challenges. In addition, different PUFs produce different responses when presented with the same input challenge.

Definition 2. Suppose t_1 and t_2 are different time periods for which we want to measure the reproducibility of a given PUF response under diverse operating conditions when presented with a specific challenge. Then PUF reliability is expressed as,

$$f_{PUF}(C_A)\Big|_{t=t_1} = f_{PUF}(C_A)\Big|_{t=t_2} \tag{5}$$

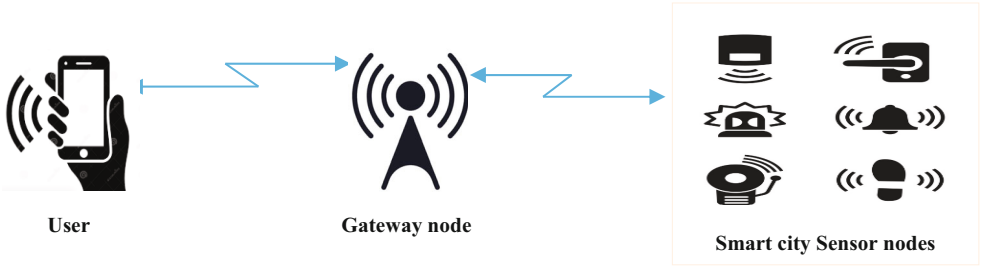

FIGURE 1 | Network model.

However, PUF responses are sensitive to diverse noise elements such as environmental changes, voltage and temperature. When presented with the same challenge, this noise may cause the same PUF to yield erroneous response that is different from the original one.

3.1.2 | Fuzzy Extraction

A fuzzy extractor utilizes user biometric data to securely execute authentication. This is a two-step process that involves probabilistic generation function *Gen* (.) and deterministic reproduction function *Rep* (.). During this process, the following definitions hold.

Definition 3. When presented with user biometric data $\mathcal{B}_i$ as an input, the *Gen* (.) function creates biometric secret $\alpha_i \in \{0,1\}^l$ together with public reproduction parameter $\beta_i \in \{0,1\}^*$ such that,

$$Gen\left(\mathcal{B}_i\right) = \left\{\alpha_i, \beta_i\right\} \tag{6}$$

Definition 4. Suppose that the deterministic reproduction function *Rep* (.) is presented with noisy biometric data $\mathcal{B}_i^*$ as input. Then, *Rep* (.) deploys the public reproduction parameter β_i associated with $\mathcal{B}_i$ to re-generate α_i as follows.

$$Rep\left(\mathcal{B}_i^*, \beta_i\right) = \alpha_i \tag{7}$$

The reproduction of α_i requires that the Hamming distance between originally registered user biometric data $\mathcal{B}_i$ and the present noisy biometric data $\mathcal{B}_i^*$ is less than or equal to some threshold error tolerance $\Delta\mathcal{B}$.

As shown in Figure 1, the network model of our scheme comprises of user U_i who deploys mobile device MD_i to access data from sensor node SN_i via the gateway node GW_j.

The main phases that are executed in our scheme include the system setup, registration, authentication, key negotiation, and parameter update. Table 2 presents the notations used throughout this paper.

The specific details of these phases are described in the subsections below.

3.2 | System Setup Phase

The goal of this phase is to have the GW_j generate its own security tokens as well as the sensor node tokens to be deployed in the subsequent phases of our scheme. Following two steps are carried out during this phase.

TABLE 2 | Notations.

Symbol	Description
GW_j	Gateway node *j* and
SN_i	Sensor node *i*
ϕ_{GW}	Master key for GW_j
SID_i	Unique identity for SN_i
R_i	Random nonce *i*
PW_i	User's password
UID_i	User's unique identity
MD_i	User mobile device
B_i	User biometric data
ΔB	Threshold error tolerance
T_i	Timestamp *i*
ΔT	Permissible transmission latency
SK_{SU}	Session key between SN_i and U_i, derived at the SN_i
SK_{US}	Session key between SN_i and U_i, derived at the MD_i
h (.)	One-way hashing function
||	Concatenation operation
$\oplus$	XOR operation

Step 1: The GW_j chooses some prime field Z_q and private master key $\phi_{GW} \in Z_q$. Next, it selects SID_i as unique identity for sensor node SN_i. This is followed by the transmission of SID_i to sensor node SN_i as shown in Figure 2. Meanwhile, the GW_j securely stores SID_i in its database and publishes *Gen* () and *PUF* ().

Step 2: Upon receiving SID_i from the GW_j, the SN_i securely stores it in its memory. Afterwards, this sensor node is ready for the registration procedures across secure channels as described below.

3.3 | Sensor Node Registration Phase

In this phase, the sensor node SN_i registers at the GW_j and gets assigned security credentials to be utilized during the authentication procedures. To accomplish this, the following four steps are performed.

Step 1: The SN_i generates random nonce R_1 and transmits registration request message $Req_1 = \{R_1, SID_i^*\}$ over to the GW_j across secure communication channels as shown in Figure 2.

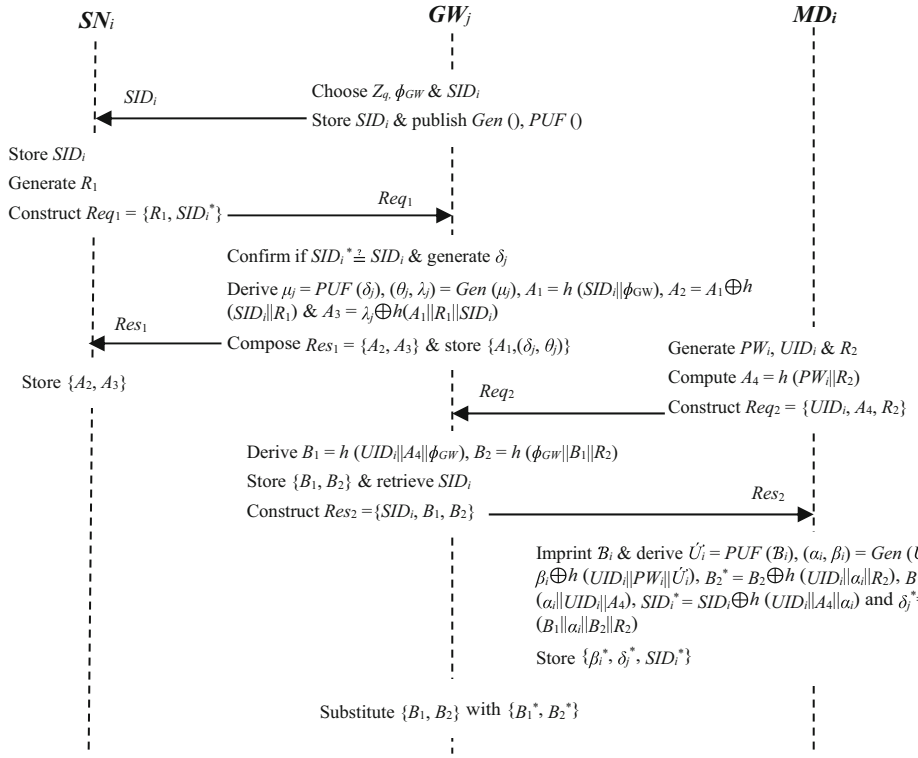

FIGURE 2 | System setup and registration phase.

Step 2: After receiving registration request message Req_1 from the SN_i, the GW_j checks if SID_i *$\stackrel{?}{=}$ SID_i. Provided that these two parameters are dissimilar, the registration process is terminated and the SN_i is prompted to re-submit valid unique identity. Otherwise, the GW_j generates random challenge set δ_j and proceeds to derive response set $\mu_j = PUF\ (\delta_j)$ for this particular δ_j.

Step 3: The GW_j computes sets θ_j, λ_j with the help of the Gen () function as $(\theta_j, \lambda_j) = Gen\ (\mu_j)$. This is followed by the computation of $A_1 = h\ (SID_i||\phi_{GW})$, $A_2 = A_1 \oplus h\ (SID_i||R_1)$ and $A_3 = \lambda_j \oplus h(A_1||R_1||SID_i)$. It then composes and sends registration response message $Res_1 = \{A_2, A_3\}$ to the SN_i over secured channels. At the end, it stores values $\{A_1, (\delta_j, \theta_j)\}$ in its database.

Step 4: On getting registration response message Res_1 from the GW_j, the SN_i stores parameters $\{A_2, A_3\}$ in its memory. This sensor node is now ready for deployment in the field of interest.

3.4 | User Registration Phase

The essence of this phase is for user U_i to submit some secret parameters to the GW_j, which in turn issues some security token to the U_i. This process is facilitated by the user mobile device MD_i as described in the following four steps.

Step 1: The user U_i generates password PW_i and unique identity UID_i. Next, it generates random nonce R_2 that is utilized in the derivation of $A_4 = h\ (PW_i||R_2)$. This is followed by the construction of registration request message $Req_2 = \{UID_i, A_4, R_2\}$ that is forwarded to the GW_j over secure channels as shown in Figure 2.

Step 2: On receiving registration request message Req_2, the GW_j derives $B_1 = h\ (UID_i||A_4||\phi_{GW})$ and $B_2 = h\ (\phi_{GW}||B_1||R_2)$. Next, it stores parameters $\{B_1, B_2\}$ in its database. This is followed by the retrieval of the sensor node SN_i identity SID_i from its database. Finally, it composes registration response message $Res_2 = \{SID_i, B_1, B_2\}$ that is transmitted over secure channel toward the U_i.

Step 3: After obtaining registration response message Res_2, the user U_i imprints his/her biometric data $\mathcal{B}_i$ to mobile device MD_i. Next, the MD_i calculates $\acute{U}_i = PUF\ (\mathcal{B}_i)$, $(\alpha_i, \beta_i) = Gen\ (\acute{U}_i)$, $\beta_i{}^* = \beta_i \oplus h\ (UID_i||PW_i||\acute{U}_i)$, $B_2{}^* = B_2 \oplus h\ (UID_i||\alpha_i||R_2)$, $B_1{}^* = B_1 \oplus h\ (\alpha_i||UID_i||A_4)$, $SID_i{}^* = SID_i \oplus h\ (UID_i||A_4||\alpha_i)$ and $\delta_j{}^* = h\ (B_1||\alpha_i||B_2||R_2)$.

Step 4: The GW_j substitutes values $\{B_1, B_2\}$ with $\{B_1{}^*, B_2{}^*\}$. Finally, the MD_i stores parameters $\{\beta_i{}^*, \delta_j{}^*, SID_i{}^*\}$ in its memory. At this juncture, user U_i is ready for the authentication phase that is described below.

3.5 | Authentication and Key Negotiation Phase

During this process, user U_i and sensor node SN_i authenticate each other before they can exchange the collected data. In these procedures, the GW_j acts as an intermediary. After successful mutual authentication, U_i and SN_i setup a session key that will be deployed to encipher the exchanged data. To accomplish this, the following eight steps are performed.

Step 1: User U_i inputs unique identity UID_i and password PW_i into the MD_i. Next, U_i imprints his/her biometric data $\mathcal{B}_i$

into the MD_i and proceeds to derive $\acute{U}_i = PUF(\mathcal{B}_i)$, $\beta_i = \beta_i^* \oplus h(UID_i||PW_i||\acute{U}_i)$, $\alpha_i = Rep(\acute{U}_i, \beta_i)$, $A_4 = h(PW_i||R_2)$, $B_2 = B_2^* \oplus h(UID_i||\alpha_i||R_2)$, $B_1 = B_1^* \oplus h(\alpha_i||UID_i||A_4)$, $SID_i = SID_i^* \oplus h(UID_i||A_4||\alpha_i)$ and $\delta_j^* = h(B_1||\alpha_i||B_2||R_2)$.

Step 2: The MD_i determines whether $\delta_j^* \stackrel{?}{=} \delta_j$ such that the session is aborted upon verification failure. Otherwise, it generates nonce R_3 and determines current timestamp T_1. Next, it calculates $B_3 = (R_3||SID_i) \oplus h(B_1||B_2||T_1)$ and $B_4 = h(B_1||R_3||B_2||T_1)$. At the end, it constructs authentication message $Auth_1 = \{B_1, B_3, B_4, T_1\}$ that is transmitted over to the GW_j across public channels as shown in Figure 3.

Step 3: After receiving authentication message $Auth_1$, the GW_j determines the current timestamp T_2 and checks whether $|T_2 - T_1| \leq \Delta T$. If this verification fails, message $Auth_1$ is flagged as a replay and the session is terminated. Otherwise, the GW_j derives $(R_3||SID_i) = B_3 \oplus h(B_1||B_2||T_1)$ and $B_4^* = h(B_1||R_3||B_2||T_1)$. Thereafter, the GW_j establishes whether $B_4^* \stackrel{?}{=} B_4$. Basically, the session is terminated if this condition does not hold. Otherwise, the GW_j retrieves $(\delta_j, \theta_j) \leftarrow SID_i$ and chooses random nonce R_4.

Step 4: The GW_j derives $A_1 = h(SID_i||\phi_{GW})$, $C_1 = (R_3||R_4) \oplus h(SID_i||\theta_j||A_1||T_2)$ and $C_2 = h(SID_i||R_4||A_1||\theta_j||T_2)$. Afterwards, it composes authentication message $Auth_2 = \{\delta_j, C_1, C_2, T_2\}$ which is forwarded to the SN_i over public channels.

Step 5: Upon getting message $Auth_2$, sensor node SN_i determines the current timestamp T_3. Next, it checks whether $|T_3 - T_2| \leq \Delta T$

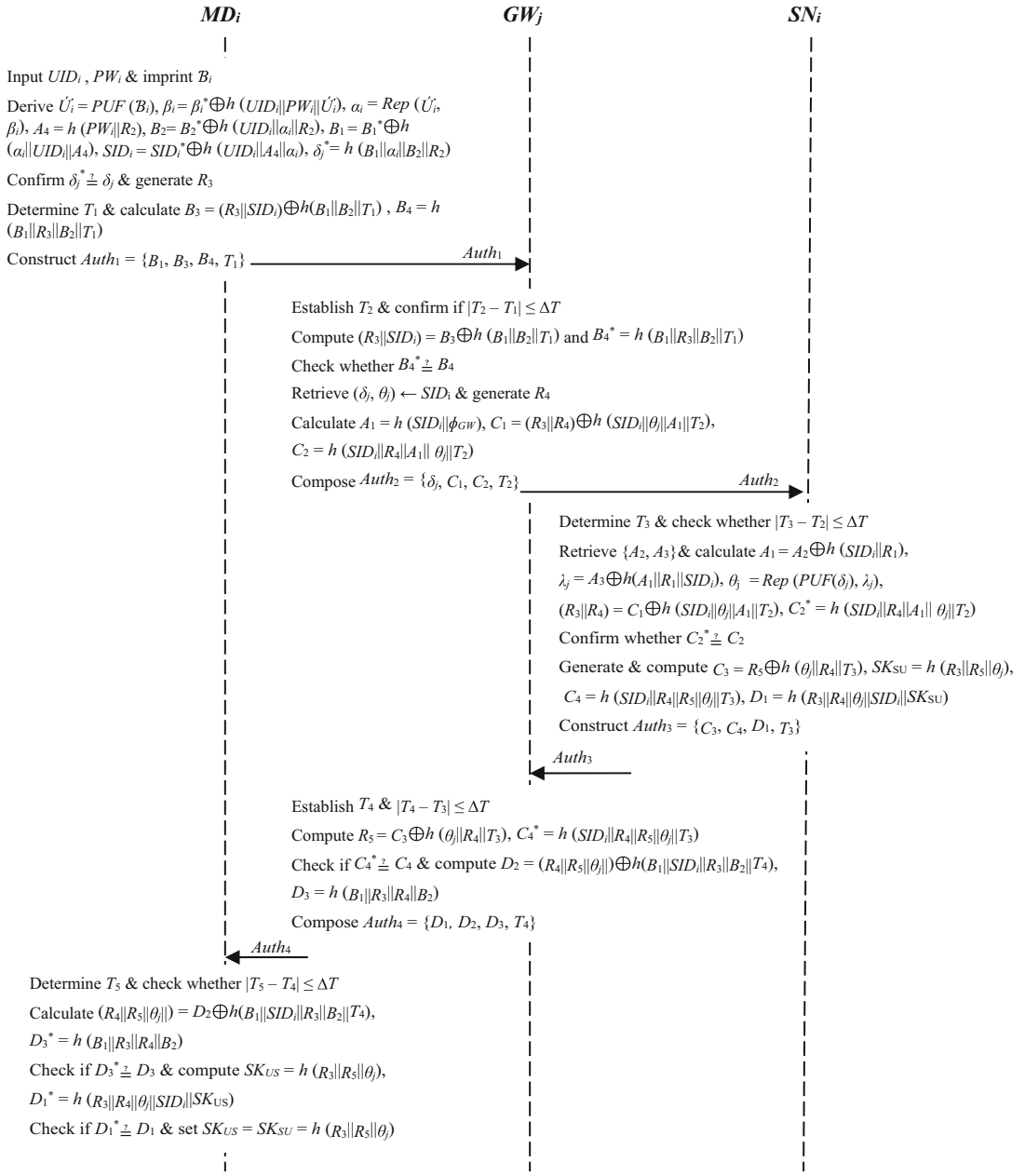

FIGURE 3 | Authentication and key negotiation phase.

such that the session is aborted upon verification failure. Otherwise, the SN_i retrieves $\{A_2, A_3\}$ from its memory and derives $A_1 = A_2 \oplus h(SID_i||R_1)$ and $\lambda_j = A_3 \oplus h(A_1||R_1||SID_i)$.

Step 6: The SN_i derives $\theta_j = Rep(PUF(\delta_j), \lambda_j)$, $(R_3||R_4) = C_1 \oplus h(SID_i||\theta_j||A_1||T_2)$ and $C_2{}^* = h(SID_i||R_4||A_1||\theta_j||T_2)$. Next, it validates whether $C_2{}^* \stackrel{?}{=} C_2$ such that the authentication process is terminated upon verification failure. Otherwise, the SN_i generates nonce R_5 and derives $C_3 = R_5 \oplus h(\theta_j||R_4||T_3)$, session key $SK_{SU} = h(R_3||R_5||\theta_j)$, $C_4 = h(SID_i||R_4||R_5||\theta_j||T_3)$, and $D_1 = h(R_3||R_4||\theta_j||SID_i||SK_{SU})$. At the end, it constructs authentication message $Auth_3 = \{C_3, C_4, D_1, T_3\}$ that is transmitted over to the GW_j across public channels.

Step 7: On receiving message $Auth_3$, the GW_j determines the current timestamp T_4 and establishes whether $|T_4 - T_3| \leq \Delta T$. Essentially, message $Auth_3$ is flagged as a replay if this verification flops and hence the session is aborted. Otherwise, the GW_j derives $R_5 = C_3 \oplus h(\theta_j||R_4||T_3)$ and $C_4{}^* = h(SID_i||R_4||R_5||\theta_j||T_3)$. It then determines whether $C_4{}^* \stackrel{?}{=} C_4$ such that the session is terminated upon validation failure.

Otherwise, the GW_j derives $D_2 = (R_4||R_5||\theta_j||) \oplus h(B_1||SID_i||R_3||B_2||T_4)$ and $D_3 = h(B_1||R_3||R_4||B_2)$. At the end, it composes authentication message $Auth_4 = \{D_1, D_2, D_3, T_4\}$ which is forwarded to U_i over public channels.

Step 8: Upon getting authentication message $Auth_4$, the U_i establishes current timestamp T_5 and checks whether $|T_5 - T_4| \leq \Delta T$. If this condition does not hold, the session is aborted. Otherwise, the MD_i calculates $(R_4||R_5||\theta_j||) = D_2 \oplus h(B_1||SID_i||R_3||B_2||T_4)$ and $D_3{}^* = h(B_1||R_3||R_4||B_2)$. Thereafter, it validates whether $D_3{}^* \stackrel{?}{=} D_3$ such that the authentication process is terminated upon verification failure. Otherwise, it derives session key $SK_{US} = h(R_3||R_5||\theta_j)$ and $D_1{}^* = h(R_3||R_4||\theta_j||SID_i||SK_{US})$. It then confirms whether $D_1{}^* \stackrel{?}{=} D_1$ such that the session is aborted if this validation flops. Otherwise, U_i and SN_i have successfully authenticated each other and established session key $SK_{US} = SK_{SU} = h(R_3||R_5||\theta_j)$.

3.6 | Parameter Change Phase

In this phase, user U_i refreshes his/her biometric data as well as password when the need for doing so arises. To reduce the computation and communication overheads, this process is executed devoid of the GW_j involvement. To achieve this, the following four steps are executed.

Step 1: User U_i inputs old password $PW_i{}^o$ and unique identity UID_i into the mobile device MD_i. This is followed by the imprint of user's old biometric data $\mathcal{B}_i{}^o$ into the MD_i.

Step 2: The MD_i derives parameters $\acute{U}_i = PUF(\mathcal{B}_i{}^o)$, $(\alpha_i, \beta_i) = Gen(\acute{U}_i)$, $\beta_i{}^* = \beta_i \oplus h(UID_i||PW_i{}^o||\acute{U}_i)$, $A_4 = h(PW_i{}^o||R_2)$, $B_2{}^* = B_2 \oplus h(UID_i||\alpha_i||R_2)$, $B_1{}^* = B_1 \oplus h(\alpha_i||UID_i||A_4)$, $SID_i{}^* = SID_i \oplus h(UID_i||A_4||\alpha_i)$ and $\delta_i{}^* = h(B_1||\alpha_i||B_2||R_2)$.

Step 3: MD_i confirms if $\delta_i{}^* \stackrel{?}{=} \delta_i$ so that the session is aborted if these values are not equivalent. Otherwise, the MD_i prompts user U_i to input new password $PW_i{}^{new}$ and imprint new biometric $\mathcal{B}_i{}^{new}$. Afterwards, the MD_i computes $\acute{U}_i{}^{new} = PUF(\mathcal{B}_i{}^{new})$, $(\alpha_i{}^{new}, \beta_i{}^{new}) = Gen(\acute{U}_i{}^{new})$, $\beta_i{}^{new^*} = \beta_i{}^{new} \oplus h(UID_i||PW_i{}^{new}||\acute{U}_i{}^{new})$, $A_4{}^{new} = h(PW_i{}^{new}||R_2)$, $B_2{}^{new^*} = B_2 \oplus h(UID_i||\alpha_i{}^{new}||R_2)$, $B_1{}^{new^*} = B_1 \oplus h(\alpha_i{}^{new}||UID_i||A_4{}^{new})$, $SID_i{}^{new^*} = SID_i \oplus h(UID_i||A_4{}^{new}||\alpha_i{}^{new})$ and $\delta_i{}^{new^*} = h(B_1||\alpha_i{}^{new}||B_2||R_2)$.

Step 4: The MD_i substitutes parameters $\{B_1{}^*, B_2{}^*, \beta_i{}^*, \delta_i{}^*, SID_i{}^*\}$ with their updates versions $\{B_1{}^{new^*}, B_2{}^{new^*}, \beta_i{}^{new^*}, \delta_i{}^{new^*}, SID_i{}^{new^*}\}$ in its memory.

4 | Security Evaluation

In this section, we present formal as well as informal security analyses of our scheme.

4.1 | Formal Security Analysis

The goal of this subsection is to demonstrate that negotiated session key is secure against active and active adversarial attacks. To achieve this, we deploy the Real-Or-Random (ROR) model. In the proposed scheme, three entities are involved during the setting up of the session key. These entities include the MD_i, SN_i, and the GW_i. Suppose that $\cap_{MD_i}^{t_1}$, $\cap_{GW_j}^{t_2}$, and $\cap_{SN_i}^{t_3}$ are the instances of t_1^{th} of MD_i, t_2^{th} of GW_i and t_3^{th} of SN_i, respectively. We let the adversary $\mathring{A}$ perform the *Send* (), *Corrupt* (), *Execute* (), *Test* (), and *Reveal* () queries against our ROR model. The descriptions of these queries are as follows.

$Send(\cap^t, Msg)$: $\mathring{A}$ transmits message Msg toward $\cap^t$ and obtains some response message.

$Corrupt_{MD_i}\left(\cap_{MD_i}^{t_1}\right)$: $\mathring{A}$ uses this query to extract private security tokens stored in the memory of MD_i. Therefore, these are basically the stolen mobile device and side-channeling attacks against ROR model.

$Corrupt_{SN_i}\left(\cap_{SN_i}^{t_2}\right)$: This query is deployed by $\mathring{A}$ to extract all security tokens stored in sensor node SN_i. This query therefore represents a physical capture attack against the ROR model.

$Execute\left(\cap_{MD_i}^{t_1}, \cap_{GW_j}^{t_2}, \cap_{SN_i}^{t_3}\right)$: The goal of this query is to enable $\mathring{A}$ eavesdrop secret credentials transmitted over the public channels. Using the captured security values, adversary $\mathring{A}$ is able to launch numerous active and passive attacks.

$Test\left(\cap^t\right)$: In this query, we consider some unbiased coin φ is considered. At the onset of the various adversarial games against ROR model, $\mathring{A}$ tosses φ. Given that condition $\varphi = 1$ is obtained, then session key $SK_{SU} = SK_{US}$ negotiated between $\cap_{MD_i}^{t_1}$ and $\cap_{SN_i}^{t_3}$ is fresh. On the other hand, if the tossing yields condition $\varphi = 0$, the session key $SK_{SU} = SK_{US}$ is not fresh. Otherwise, the adversarial tossing yields a null value ($\perp$).

$Reveal\left(\cap^t\right)$: Using this query, $\mathring{A}$ discloses session key $SK_{SU} = SK_{US}$ negotiated between $\cap_{MD_i}^{t_1}$ and $\cap_{SN_i}^{t_3}$.

In addition to the above queries, *PUF* () and one-way hash function *Hash* () random oracles are utilized.

Theorem. *Denoting our PUF and biometric based scheme as (Pb), then $Adv_{Å}^{Pb}$ represents the advantage of Å in compromising session key $SK_{SU} = SK_{US}$ negotiated in our scheme. Taking N_S, N_H and N_P as the number of Send (), Hash () and PUF () queries, respectively, $|\mathbb{H}|$ and $|\mathbb{P}|$ as the range space of hash () and PUF () queries, respectively, L_B as the bit length in biometric secret key and L_P as the bit length in* PUF *secret, $\acute{Z}_1$ and $\acute{Z}_2$ as the parameters deployed in Zipf's law, then we obtain the following.*

$$Adv_{Å}^{Pb} \leq \frac{N_H^2}{|\mathbb{H}|} + \frac{N_P^2}{|\mathbb{P}|} + 2\max\left\{Z_2.N_S^{Z_1}, \frac{N_S}{2^{L_B}}, \frac{N_S}{2^{L_P}}\right\}$$

Proof. We formulate a sequence of games Gm_i ($j \in [0,4]$) to demonstrate the robustness of the negotiated session key $SK_{SU} = SK_{US} = h\ (R_3||R_5||\theta_i)$. In all these games, the probability of Å winning game Gm_i is denoted as $Adv_{Å,Gm_j}^{Pb}$. The specific details of these games are described below.

Gm_0: This is the actual attacks that are launched by Å against our scheme. Here, a bit φ is randomly generated prior to Gm_0 and hence,

$$Adv_{Å}^{Pb} = |2.Adv_{Å,Gm_0}^{Pb} - 1 \tag{8}$$

Gm_1: In this game, adversary Å performs eavesdropping attacks by intercepting the messages $Auth_1$, $Auth_2$, $Auth_3$, and $Auth_4$ exchanged among the MD_i, SN_i, and the GW_i. To achieve this, an *Execute* () query is launched by Å. Thereafter, *Reveal* () and *Test* () queries are performed so as to derive session key SK_{SU} or SK_{US}. By using the outputs of the *Reveal* () and *Test* () queries, it can be determined whether Å has managed to obtain the private security tokens and session key $SK_{US} = SK_{SU} = h\ (R_3||R_5||\theta_i)$ negotiated between the MD_i and SN_i. To derive a valid session key, Å needs access to the PUF private parameter θ_i as well as random nonces $\{R_3, R_5\}$. Since these parameters are never exchanged in plaintext over the public channels, they cannot be intercepted by the adversary. Therefore, the probability of Å winning Gm_1 via eavesdropping has not increased. As such,

$$Adv_{Å,Gm_1}^{Pb} = Adv_{Å,Gm_0}^{Pb} \tag{9}$$

Gm_2: In this game, the *Send* () and *Hash* () queries are deployed by to launch active and passive attacks against our protocol. To begin with, Å eavesdrops messages $Auth_1 = \{B_1, B_3, B_4, T_1\}$, $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$, $Auth_3 = \{C_3, C_4, D_1, T_3\}$, and $Auth_4 = \{D_1, D_2, D_3, T_4\}$ exchanged during the authentication and key agreement process. Here, $B_1 = B_1^* \oplus h\ (\alpha_i||UID_i||A_4)$, $B_3 = (R_3||SID_i) \oplus h(B_1||B_2||T_1)$, $B_4 = h\ (B_1||R_3||B_2||T_1)$, $C_1 = (R_3||R_4) \oplus h\ (SID_i||\theta_i||A_1||T_2)$, $C_2 = h\ (SID_i||R_4||A_1||\ \theta_i||T_2)$, $C_3 = R_5 \oplus h\ (\theta_i||R_4||T_3)$, $C_4 = h\ (SID_i||R_4||R_5||\theta_i||T_3)$, $D_1 = h\ (R_3||R_4||\theta_i||SID_i||SK_{SU})$, $D_2 = (R_4||R_5||\theta_i||) \oplus h(B_1||SID_i||R_3||B_2||T_4)$, and $D_3 = h\ (B_1||R_3||R_4||B_2)$. Evidently, all these messages are protected by one-way hash function h (). As such, random nonces R_3, R_4, and R_5 can never be revealed from the intercepted messages. On the basis of the birthday paradox, the following is obtained.

$$|Adv_{Å,Gm_2}^{Pb} - Adv_{Å,Gm_1}^{Pb}| \leq \frac{N_H^2}{2|\mathbb{H}|} \tag{10}$$

Gm_3: This game involves the simulation of the *PUF* () query. On the basis of the arguments in Gm_2, and the security of PUF functions described in the *Mathematical preliminaries* section, we obtain the following.

$$|Adv_{Å,Gm_3}^{Pb} - Adv_{Å,Gm_2}^{Pb}| \leq \frac{N_P^2}{2|\mathbb{P}|} \tag{11}$$

Gm_4: This is the final game in which adversary Å simulates $Corrupt_{MD_i}$ and $Corrupt_{SN_i}$ queries. It is assumed that parameters $\{\beta_i^*, \delta_i^*, SID_i^*\}$ stored in MD_i memory can be extracted via power analysis. Similarly, values $\{A_2, A_3\}$ can be retrieved through physical node capture attacks. Here, $\beta_i^* = \beta_i \oplus h\ (UID_i||PW_i||\acute{U}_i)$, $\delta_i^* = h\ (B_1||\alpha_i||B_2||R_2)$, $SID_i^* = SID_i \oplus h\ (UID_i||A_4||\alpha_i)$, $A_2 = A_1 \oplus h\ (SID_i||R_1)$, and $A_3 = \lambda_i \oplus h(A_1||R_1||SID_i)$. However, it is still computationally infeasible to obtain user password PW_i through the *Send*() query devoid of the user real identity U_i, nonce R_2 and PUF secret value α_i. In addition, Å is unable to distinguish between PUF secret value and biometric secret parameter. This is because the probabilities of correctly guessing biometric secret of L_B bits is $1/2^{L_B}$ while that of correctly guessing PUF secret value of L_P bits is $1/2^{L_P}$. Therefore, Gm_4 and Gm_3 are indistinguishable devoid of successful biometric guessing and offline password guessing attacks. As such,

$$|Adv_{Å,Gm_4}^{Pb} - Adv_{Å,Gm_3}^{Pb}| \leq \max\left\{Z_2.N_S^{Z_1}, \frac{N_S}{2^{L_B}}, \frac{N_S}{2^{L_P}}\right\} \tag{12}$$

Finally, adversary Å attempts to guess correct bit φ in an attempt to win the game. To accomplish this, the *Test* () query is performed. As such, the following outcome is obtained.

$$Adv_{Å,Gm_4}^{Pb} = \frac{1}{2} \tag{13}$$

The amalgamation of relations (8), (9), and (13) yields the following.

$$\begin{aligned}\frac{1}{2}Adv_{Å}^{Pb} &= \left|Adv_{Å,Gm_0}^{Pb} - \frac{1}{2}\right| \\ &= \left|Adv_{Å,Gm_1}^{Pb} - \frac{1}{2}\right| \\ &= \left|Adv_{Å,Gm_1}^{Pb} - |Adv_{Å,Gm_4}^{Pb}\right| \end{aligned} \tag{14}$$

The application of the triangular inequality in (10–12) and (14) yields the following:

$$\begin{aligned}\frac{1}{2}Adv_{Å}^{Pb} &= |Adv_{Å,Gm_1}^{Pb} - Adv_{Å,Gm_4}^{Pb}| \\ &\leq |Adv_{Å,Gm_1}^{Pb} - Adv_{Å,Gm_3}^{Pb}| \\ &\quad + |Adv_{Å,Gm_3}^{Pb} - Adv_{Å,Gm_4}^{Pb}| \\ &\leq |Adv_{Å,Gm_1}^{Pb} - Adv_{Å,Gm_2}^{Pb}| \\ &\quad + |Adv_{Å,Gm_2}^{Pb} - Adv_{Å,Gm_3}^{Pb}| \\ &\quad + |Adv_{Å,Gm_3}^{Pb} - Adv_{Å,Gm_4}^{Pb}| \\ &\leq \frac{N_H^2}{2|\mathbb{H}|} + \frac{N_P^2}{2|\mathbb{P}|} + \max\left\{Z_2.N_S^{Z_1}, \frac{N_S}{2^{L_B}}, \frac{N_S}{2^{L_P}}\right\} \end{aligned} \tag{15}$$

The multiplication of both sides of relation (15) above by 2 yields the following.

$$Adv_{\text{Å}}^{Pb} \leq \frac{N_H^2}{|\mathbb{H}|} + \frac{N_P^2}{|\mathbb{P}|} + 2\max\left\{Z_2.N_S^{Z_1}, \frac{N_S}{2^{L_B}}, \frac{N_S}{2^{L_P}}\right\} \quad (16)$$

Relation (16) above completes the proof and hence our scheme is robust against passive and active attacks. □

4.2 | Informal Security Analysis

In this subsection, we state and proof various lemmas to demonstrate the resilience of our scheme against a wide range of smart city attack vectors.

Lemma 1. *Strong mutual authentication is attained.*

Proof. In the proposed protocol, the MD_i, GW_i and U_i mutually authenticate each other before accepting each other's messages. For instance, on receiving message $Auth_1 = \{B_1, B_3, B_4, T_1\}$ from MD_i, the GW_i derives $B_4{}^* = h\ (B_1||R_3||B_2||T_1)$ and checks whether $B_4{}^* \stackrel{?}{=} B_4$. Similarly, upon getting message $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$ from GW_i, the SN_i computes $C_2{}^* = h\ (SID_i||R_4||A_1||\theta_i||T_2)$ and confirms if $C_2{}^* \stackrel{?}{=} C_2$. On the other hand, on getting message $Auth_3 = \{C_3, C_4, D_1, T_3\}$ from SN_i, the GW_i calculates $C_4{}^* = h\ (SID_i||R_4||R_5||\theta_i||T_3)$ and determines whether $C_4{}^* \stackrel{?}{=} C_4$. Similarly, U_i receives message $Auth_4 = \{D_1, D_2, D_3, T_4\}$ from GW_i after which it derives $D_3{}^* = h\ (B_1||R_3||R_4||B_2)$ and confirms if $D_3{}^* \stackrel{?}{=} D_3$. In all these validations, the authentication session is terminated upon validation failure. □

Lemma 2. *Our protocol prevents impersonation and eavesdropping attacks.*

Proof. Suppose that adversary $\mathring{A}$ eavesdrops the authentication messages $Auth_1 = \{B_1, B_3, B_4, T_1\}$, $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$, $Auth_3 = \{C_3, C_4, D_1, T_3\}$, and $Auth_4 = \{D_1, D_2, D_3, T_4\}$ exchanged over the public channels. Thereafter, attempts are made to impersonate MD_i, GW_i and U_i by generating valid requests and response messages. Here, $A_4 = h\ (PW_i||R_2)$, $B_1 = B_1{}^* \oplus h\ (\alpha_i||UID_i||A_4)$, $B_2 = h\ (\Phi_{GW}||B_1||R_2)$, $B_3 = (R_3||SID_i) \oplus h(B_1||B_2||T_1)$, $B_4 = h\ (B_1||R_3||B_2||T_1)$, $C_1 = (R_3||R_4) \oplus h\ (SID_i||\theta_i||A_1||T_2)$, $C_2 = h\ (SID_i||R_4||A_1||\theta_i||T_2)$, $C_3 = R_5 \oplus h\ (\theta_i||R_4||T_3)$, $C_4 = h\ (SID_i||R_4||R_5||\theta_i||T_3)$, $D_1 = h\ (R_3||R_4||\theta_i||SID_i||SK_{SU})$, $D_2 = (R_4||R_5||\theta_i||) \oplus h(B_1||SID_i||R_3||B_2||T_4)$, and $D_3 = h\ (B_1||R_3||R_4||B_2)$. Clearly, to compose any valid authentication messages, random nonces R_2, R_3, R_4, and R_5 are required. In addition, unique identity SID_i for SN_i, session key SK_{SU}, master key Φ_{GW} for GW_i, user password PW_i and identity UID_i are all needed. Since these values are never exchanged in plaintext over the public internet, they cannot be obtained by $\mathring{A}$ and hence these two attacks flop. □

Lemma 3. *Anonymity and untraceability are preserved.*

Proof. During the authentication and key negotiation phase, messages $Auth_1 = \{B_1, B_3, B_4, T_1\}$, $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$, $Auth_3 = \{C_3, C_4, D_1, T_3\}$, and $Auth_4 = \{D_1, D_2, D_3, T_4\}$ exchanged over the public internet. Based on Lemma 2 above, none of these messages contain the real identity SID_i of the sensor node or the real identity UID_i of the user. Although parameters $B_1 = B_1{}^* \oplus h\ (\alpha_i||UID_i||A_4)$, $B_3 = (R_3||SID_i) \oplus h(B_1||B_2||T_1)$, $C_1 = (R_3||R_4) \oplus h\ (SID_i||\theta_i||A_1||T_2)$, $C_2 = h\ (SID_i||R_4||A_1||\theta_i||T_2)$, $C_4 = h\ (SID_i||R_4||R_5||\theta_i||T_3)$, $D_1 = h\ (R_3||R_4||\theta_i||SID_i||SK_{SU})$, and $D_2 = (R_4||R_5||\theta_i||) \oplus h(B_1||SID_i||R_3||B_2||T_4)$ contain these identities, they are masked in other parameters and protected by the one-way hashing function. As such, $\mathring{A}$ cannot easily extract these real identities from the exchanged messages and therefore anonymity is upheld. Consequently, an attacker is unable to relate these messages to the specific entity and hence untraceability of the communication process is preserved. □

Lemma 4. *Replay attacks are prevented.*

Proof. The assumption made in these attacks is that $\mathring{A}$ has eavesdropped messages $Auth_1$, $Auth_2$, $Auth_3$, and $Auth_4$ exchanged among the MD_i, SN_i, and the GW_i. According to Lemma 2, all these messages incorporate timestamps whose freshness is verified at the receiver end. For instance, on getting message $Auth_1 = \{B_1, B_3, B_4, T_1\}$ from MD_i, the GW_i determines the current timestamp T_2 and checks whether $|T_2 - T_1| \leq \Delta T$. Similarly, upon receiving message $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$ from GW_i, the SN_i determines the current timestamp T_3 and checks whether $|T_3 - T_2| \leq \Delta T$. On the other hand, the GW_i determines the current timestamp T_4 and establishes whether $|T_4 - T_3| \leq \Delta T$ upon receiving message $Auth_3 = \{C_3, C_4, D_1, T_3\}$. Similarly, the U_i establishes current timestamp T_5 and checks whether $|T_5 - T_4| \leq \Delta T$ upon getting authentication message $Auth_4 = \{D_1, D_2, D_3, T_4\}$. In all these cases, the authentication messages are flagged as replay when these verifications flop. In addition, random nonces R_2, R_3, R_4, and R_5 are incorporated in these messages to preserve their freshness. □

Lemma 5. *Privileged insider and Ephemeral Secret Leakage (ESL) attacks are thwarted.*

Proof. Suppose that some privileged insider has captured long terms secrets such as master Φ_{GW}. Thereafter, attempts are made to compromise the derived session key $SK_{US} = SK_{SU} = h\ (R_3||R_5||\theta_i)$. Here, $R_5 = C_3 \oplus h\ (\theta_i||R_4||T_3)$ and $\theta_i = Rep\ (PUF(\delta_i), \lambda_i)$. Since δ_i and λ_i are unavailable to $\mathring{A}$, this attack flops. Similarly, we assume that short-term secrets such as nonces R_3 and R_5 have been compromised by the privileged insider. Still, $\mathring{A}$ is unable to derive session key $SK_{US} = SK_{SU} = h\ (R_3||R_5||\theta_i)$ due to lack of δ_i and λ_i, hence these two attacks flop. □

Lemma 6. *Offline password guessing attacks are thwarted.*

Proof. In this attack, it is assumed that messages $Auth_1$, $Auth_2$, $Auth_3$ and $Auth_4$ have been intercepted. In addition, we assume that the secret credentials $\{\beta_i{}^*, \delta_i{}^*, SID_i{}^*\}$ stored in MD_i memory have been retrieved by $\mathring{A}$. Thereafter, an offline password guessing attack is launched against PW_i. According to Lemma 2, PW_i is only incorporated in parameter $A_4 = h\ (PW_i||R_2)$. Due to the incapsulation in nonce R_2 and one-way hashing function, it is computationally infeasible for $\mathring{A}$ to guess password PW_i. Regarding the memory resident parameters, $\beta_i{}^* = \beta_i \oplus h\ (UID_i||PW_i||\acute{U}_i)$, $\delta_i{}^* = h\ (B_1||\alpha_i||B_2||R_2)$, $SID_i{}^* = SID_i \oplus h\ (UID_i||A_4||\alpha_i)$. Due to the difficulty of reversing the one-way hashing function, $\mathring{A}$ cannot retrieve A_4. Therefore, offline guessing attacks against our scheme has flopped. □

Lemma 7. *Our protocol is robust against physical and side-channeling attacks.*

Proof. In these attacks, the adversary $\mathring{A}$ is deemed to have the capability of physically capturing sensor node SN_i. Thereafter, values $\{A_2, A_3\}$ stored in SN_i are extracted through power analysis. Here, $A_1 = h\ (SID_i||\Phi_{GW})$, $A_2 = A_1 \oplus h\ (SID_i||R_1)$ and $A_3 = \lambda_i \oplus h(A_1||R_1||SID_i)$. Next, attempts are made to derive the shared session key $SK_{US} = SK_{SU} = h\ (R_3||R_5||\theta_i)$. Evidently, this derivation will fail since $\mathring{A}$ has no access to random nonces R_3 and R_5. In addition, parameter $\theta_i = Rep\ (PUF(\delta_i),\ \lambda_i)$ is unavailable to $\mathring{A}$. Since PUF challenge and response pair $\{\delta_i,\ \theta_i\}$ are distinct and independent for each sensor node, their compromise does not necessarily lead to the compromise of other sensor node SN_k. □

Lemma 8. *MitM and session hijacking attacks are prevented.*

Proof. Suppose that authentication messages $Auth_1$, $Auth_2$, $Auth_3$ and $Auth_4$ have been intercepted by $\mathring{A}$. Thereafter, adversary $\mathring{A}$ tries to compose and insert bogus messages ${Auth_1}^b$, ${Auth_2}^b$, ${Auth_3}^b$, and ${Auth_4}^b$ into the communication channel. According to Lemma 2, random nonces $\{R_2, R_3, R_4, R_5\}$ are needed for any successful derivation of these messages. In addition, secrets SID_i, SK_{SU}, Φ_{GW}, PW_i, and UID_i are all required. Since these parameters are unavailable to $\mathring{A}$, this attack flops. Let us assume that $\mathring{A}$ is interested in deriving session key $SK_{US} = SK_{SU} = h\ (R_3||R_5||\theta_i)$ negotiated between SN_i and MD_i. Afterwards, attempts are made to hijack the communication session between SN_i and MD_i. However, devoid of PUF parameter θ_i and nonces R_3 and R_5, this derivation fails. As such, both MitM and session hijacking attacks against our protocol have flopped. □

Lemma 9. *Our scheme can withstand session key disclosure and forgery attack.*

Proof. In this attack, we assume that user U_i has lost his/her mobile device MD_i. It is also assumed that the current session key has been captured by $\mathring{A}$. Therefore, parameters $\{\beta_i^*,\ \delta_i^*,\ SID_i^*\}$ stored in MD_i memory are extracted. Thereafter, an adversary may try to forge session key $SK_{SU} = h\ (R_3||R_5||\theta_i)$ for the subsequent communication procedures. However, according to Lemmas 5 and 8, the adversary lacks the secret credentials required to compute the session key. Since $\theta_i = Rep\ (PUF(\delta_i), \lambda_i)$, adversary $\mathring{A}$ lacks PUF secret parameter λ_i required to derive θ_i. In addition, the keying parameters R_3, R_5 and θ_i cannot be retrieved from the captured session key due to the collision-resistant one way hashing function. Therefore, these two attacks cannot be successful against our scheme. □

Lemma 10. *Session key is negotiated for payload enciphering.*

Proof. In our protocol, the MD_i and SN_i setup session keys for the protection of their payloads exchanged over the public channels. Upon receiving authentication message $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$ from GW_i, the SN_i validates it by checking whether $|T_3 - T_2| \leq \Delta T$ and $C_2^* \stackrel{?}{=} C_2$. Provided that these authentications are successful, it derives the session key as $SK_{SU} = h\ (R_3||R_5||\theta_i)$. Similarly, on getting authentication message $Auth_4 = \{D_1, D_2, D_3, T_4\}$ from GW_i, the MD_i verifies it by confirming if $|T_5 - T_4| \leq \Delta T$ and $D_3^* \stackrel{?}{=} D_3$. Afterwards, it computes session key as $SK_{US} = h\ (R_3||R_5||\theta_i)$ on condition that these verifications are successful. □

Lemma 11. *Our protocol preserves key secrecy.*

Proof. Suppose that adversary has captured the current session keys $SK_{SU} = h\ (R_3||R_5||\theta_i)$ and $SK_{US} = h\ (R_3||R_5||\theta_i)$, where $\theta_i = Rep\ (PUF(\delta_i),\ \lambda_i)$. Evidently, these session keys incorporate random nonce R_3 and R_5. Since these nonces are session-specific, they cannot be reused by to derive keys for the past or subsequent sessions. In addition, parameter δ_i is refreshed as $\delta_i^* = h\ (B_1||\alpha_i||B_2||R_2)$ and therefore it is also one-time due to inclusion of nonce R_2. Consequently, our protocol preserves both backward and forward key secrecy. □

5 | Performance Evaluation

In this section, we present the comparative performance evaluation of our scheme in terms of functionalities, security features, computation, and communication overheads.

5.1 | Computation Overheads

The execution time of the various cryptographic primitives executed in our protocol are used to derive the computation overheads. For the GW_i, a 64-bit laptop running on Intel Core i5-10400 processor, Ubuntu 22.04.0 LTS operating system, 8 GB of RAM and 2.90 Ghz clock speed is deployed. On the other hand, Raspberry PI 4B model, 1.5 Ghz Quad-core processor, CPU Architecture, 8GB of RAM, running on Ubuntu 22.04.0 LTS is deployed for experimentations involving the MD_i and SN_i. Under these environments, the runtime for the various cryptographic operations are presented in Table 3.

During the authentication and key negotiation phase, the MD_i performs $12T_h$ and 1 T_{fe} operations while the SN_i executes $8T_h$ and 1 T_{fe} operations. On the other hand, the GW_i performs only $8T_h$ operations. Table 4 presents the derivation of computation overheads of our scheme as well as the derivations in other related protocols.

As shown in Figure 4, the protocol in [41] incurs the highest computation overheads of 24.167 ms. This is followed by the protocols

TABLE 3 | Cryptographic runtimes.

Operation	MD_i/SN_i	GW_i
One-way hashing, T_h	0.356	0.058
Fuzzy extraction, T_{fe}	2.862	0.547
Symmetric encryption/decryption, T_s	0.938	0.082
Physical unclonable function, T_{puf}	0.309	0.018
Elliptic curve scalar point multiplication, T_m	2.862	0.547

TABLE 4 | Computation overheads.

Protocol	MD_i	GW_i	SN_i	Total (ms)
Li et al. [24]	$13T_h + 3T_m \approx 13.214$	$8T_h + T_m \approx 1.011$	$4T_h + 2T_m \approx 7.148$	21.373
Nikooghadam et al. [32]	$6T_h + 2T_m \approx 7.86$	$8T_h \approx 0.464$	$5T_h + 2T_m \approx 7.504$	15.828
Lee, Oh, and Park [36]	$13T_h + T_{fe} \approx 7.49$	$15T_h \approx 0.87$	$6T_h + T_{fe} + T_{puf} \approx 5.307$	13.667
Yuanbing, Wanrong, and Bin [39]	$14T_h + 2T_m \approx 10.708$	$10T_h \approx 0.58$	$6T_h + 2T_m \approx 7.86$	19.148
Lu et al. [41]	$7T_h + T_{fe} + 3T_m + T_s \approx 14.878$	$6T_h + T_m + T_s \approx 0.977$	$2T_h + 2T_m + 2T_s \approx 8.312$	24.167
Mo et al. [42]	$12T_h + T_{fe} + 2T_m \approx 12.858$	$10T_h + T_s \approx 1.518$	$5T_h + 2T_m + T_s \approx 8.442$	22.818
Proposed	$12T_h + T_{fe} \approx 7.134$	$9T_h \approx 0.522$	$8T_h + T_{fe} \approx 5.71$	13.366

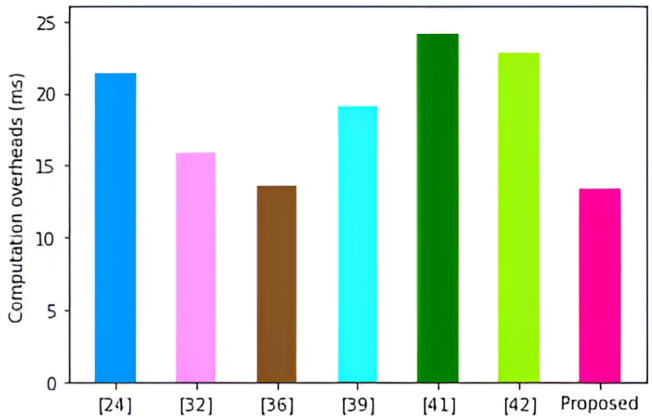

FIGURE 4 | Computation overheads comparisons [39,42].

in [24, 32, 36, 39, 42] with computation costs of 22.818, 21.373, 19.148, 15.828, and 13.667 ms, respectively.

The extensive elliptic curve scalar point multiplications in [24, 32, 36, 39, 41, 42] account for the high computation overheads. On the other hand, our scheme incurs the lowest computation costs of only 13.366 ms. Since the smart city sensors are limited in terms of computation power, our scheme is the most efficient and hence the most suitable for deployment in these sensors.

5.2 | Communication Overheads

In this section, the messages exchanged during authentication and key setup phase are deployed to derive the communication costs of our scheme. Here, we take the size of timestamp, prime number, random nonce, hash function and real identities to be 32, 160, 128, 160, and 32 bits, respectively. Using these values, the communication overheads of the various messages are derived as follows.

$\boldsymbol{MD_i \rightarrow GW_i}$: $Auth_1 = \{B_1, B_3, B_4, T_1\}$

$B_1 = B_3 = B_4 = 160$ bit, $T_1 = 32$ bits and hence $Auth_1 = 512$ bits.

$\boldsymbol{GW_i \rightarrow SN_i}$: $Auth_2 = \{\delta_i, C_1, C_2, T_2\}$

$\delta_i = C_1 = C_2 = 160$ bits, $T_2 = 32$ bits, yielding 512 bits as the size of $Auth_2$.

$\boldsymbol{SN_i \rightarrow GW_i}$: $Auth_3 = \{C_3, C_4, D_1, T_3\}$

TABLE 5 | Communication overheads.

Protocol	Size (bits)
Li et al. [24]	1792
Nikooghadam et al. [32]	2336
Lee, Oh, and Park [36]	2016
Yuanbing, Wanrong, and Bin [39]	3552
Lu et al. [41]	2880
Mo et al. [42]	3328
Proposed	2048

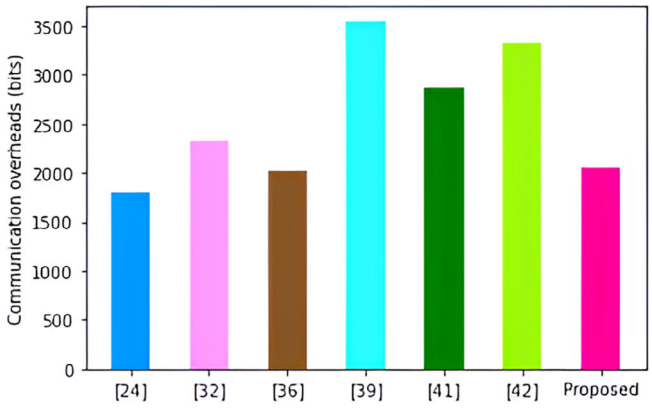

FIGURE 5 | Communication overheads comparisons.

$C_3 = C_4 = D_1 = 160$ bit, $T_2 = 32$ bits and hence the size of $Auth_3$ is 512 bits.

$\boldsymbol{GW_i \rightarrow U_i}$: $Auth_4 = \{D_1, D_2, D_3, T_4\}$

$D_1 = D_2 = D_3 = 160$ bits, $T_4 = 32$ bits, yielding 512 bits as the size of $Auth_4$.

Based on the above derivations, the cumulative communication overhead of our protocol is 2048 bits. Table 5 presents the comparisons of the derived communication cost against other related schemes.

As shown in Figure 5, the scheme in [39] incurs the highest communication costs of 3552 bits. This is followed by the protocols in [32, 41, 42], the proposed scheme, [24, 36] with communication costs of 3328, 2880, 2336, 2048, 2016, and 1792 bits, respectively.

TABLE 6 | Energy consumption.

Protocol	Energy (mJ)
Li et al. [24]	77.77
Nikooghadam et al. [32]	81.64
Lee, Oh, and Park [36]	57.74
Yuanbing, Wanrong, and Bin [39]	85.52
Lu et al. [41]	90.43
Mo et al. [42]	91.85
Proposed	62.12

Although the scheme in [24] incurs the lowest communication overhead, it cannot withstand impersonation, MitM, replay, parallel session, and privileged insider attacks. In addition, it cannot uphold forward key secrecy. Similarly, the scheme in [36] is not evaluated against attacks such as eavesdropping, ephemeral secret leakage, and side-channeling. Therefore, our scheme provides a good trade-off between bandwidth and security.

5.3 | Energy Consumption

The sensors deployed in smart cities may be deployed in locations where frequent replacement of batteries is challenging. Since these sensors are resource-constrained in terms of energy, an ideal authentication scheme should be lightweight so as to efficiently utilize the sensor energy. As such, we derive the energy consumption of our scheme and compare it with other related schemes. As discussed in [51], energy consumption is computed as follows.

$$\text{Energy} = C \times \text{P}$$

where C is the computation overhead and P is the maximum processing power. In wireless transmission systems, P is taken to be 10.88 watts [52]. Table 6 presents the energy comparisons of our protocol against other state-of-the-art schemes.

Based on the graphs in Figure 6, the sensors in the protocol developed in [42] dissipate the highest energy of 91.85 mJ. This is closely followed by the protocols in [24, 32, 39, 41], the proposed protocol, and the scheme in [36] with energy dissipation of 90.43, 85.52, 81.64, 77.77, 62.12, and 57.74 mJ, respectively.

Although the scheme in [36] dissipates the lowest energy, it has not been evaluated against attacks such as eavesdropping, ephemeral secret leakage, and side-channeling. Therefore, our scheme provides a good trade-off between energy and security.

5.4 | Functionalities and Security Features

In this section, we compare the functionalities provided by our scheme against those provided by other related techniques. In addition, the resilience of our protocol against attacks is compared with other state-of-the-art schemes as shown in Table 7.

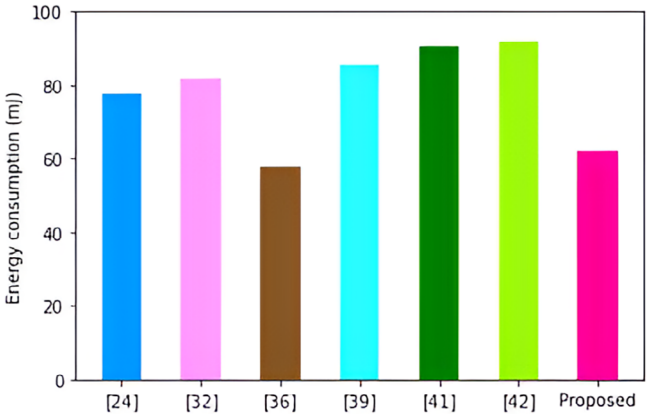

FIGURE 6 | Energy consumption comparisons.

TABLE 7 | Functionalities and security features.

	[41]	[36]	[39]	[24]	[42]	[32]	Proposed
Security features							
Mutual authentication	√	√	√	√	√	×	√
Key agreement	√	√	√	√	√	√	√
Backward key secrecy	√	√	√	×	√	√	√
Forward key secrecy	√	√	√	×	√	√	√
Anonymity	√	√	×	√	√	√	√
Untraceability	√	×	×	√	√	√	√
Password change	√	√	√	√	√	×	√
Formal verification	√	√	√	√	√	√	√
Resilient against							
Offline guessing	×	√	√	√	×	√	√
Session key disclosure	×	×	×	√	×	√	√
Eavesdropping	×	×	×	×	×	×	√
Session hijacking	×	×	×	×	×	×	√
ESL	×	√	√	×	×	×	√
Replays	√	√	√	×	√	×	√
Forgery	×	×	×	×	×	×	√
MitM	√	√	×	×	√	×	√
Privileged insider	√	×	√	×	√	×	√
Physical	√	√	×	√	×	√	√
Side-channeling	×	×	×	×	×	×	√
Impersonation	√	√	×	×	√	×	√

Note: √ Supported; × Not supported or not considered.

As shown in Table 7, the scheme in [24, 32] each support only nine security features and functionalities and hence are the most insecure. On the other hand, the protocol in [39] supports 10 functionalities and security features while the scheme in [42] offers support for 12 features. On their part, the protocols in [36, 41] support 13 features each. However, the proposed protocol supports all the 20 functionalities and security features and hence is the most secure. It has been shown that the proposed scheme incurs the lowest computation overheads, the second lowest energy consumption, and the third communication overheads. As such, it offers an ideal trade-off between security and performance and hence is the most ideal for deployment for the sensor-based smart city environment.

6 | Conclusion

Smart cities have been shown to be critical in the provision of surveillance and monitoring that enhance residents' quality of life and safety. However, vulnerabilities and heterogeneity in the deployed technologies and devices coupled with the usage of public channels for data exchanges expose these networks to numerous security and privacy threats. Although many protocols have been put in place to curb these challenges, the attainment of perfect security at optimum performance still remains quite challenging. As such, we have presented an authentication scheme using user biometrics and PUF that has been shown to set up a provably secure session key for traffic enciphering. In addition, its resilience against typical smart city attacks such as side-channeling and impersonations has been demonstrated. Moreover, comparative performance evaluations have shown that it incurs the lowest computation costs, consumes the least energy at relatively lower communication overheads. Future research work will involve the reduction of the obtained communication costs so that it can make efficient use of the network bandwidth.

Author Contributions

Vincent Omollo Nyangaresi: conceptualization, formal analysis, validation, writing – review and editing, supervision, writing – original draft. **Ahmad A. AlRababah:** methodology, software. **Ganesh Keshaorao Yenurkar:** investigation, data curation. **Ravikumar Chinthaginjala:** resources, visualization. **Muhammad Yasir:** project administration.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The authors confirm that the data supporting the findings of this study are available within the article.

References

1. S. Jiang, Q. Yu, X. Chen, L. Chen, and X. Zhang, "The Application of IoT, Big Data, and Artificial Intelligence Technologies in Smart Cities: A Review," *Journal of Sensors* 2021 (2021): 1–20.

2. V. Sucasas, A. Aly, G. Mantas, J. Rodriguez, and N. Aaraj, "Secure Multi-Party Computation-Based Privacy-Preserving Authentication for Smart Cities," *IEEE Transactions on Cloud Computing* 11, no. 4 (2023): 3555–3572.

3. J. Yang, Y. Kwon, and D. Kim, "Regional Smart City Development Focus: The South Korean National Strategic Smart City Program," *IEEE Access* 9 (2020): 7193–7210.

4. P. M. Rao, S. Pedada, S. Jangirala, A. K. Das, and J. J. Rodrigues, "Role of IoT in the Ages of Digital to Smart Cities: Security Challenges and Countermeasures," *IEEE Internet of Things Magazine* 7, no. 1 (2024): 56–64.

5. X. Hou, L. Xin, Y. Fu, et al., "A Self-Powered Biomimetic Mouse Whisker Sensor (BMWS) Aiming at Terrestrial and Space Objects Perception," *Nano Energy* 118 (2023): 109034.

6. J. Gao, D. Wu, F. Yin, Q. Kong, L. Xu, and S. Cui, "MetaLoc: Learning to Learn Wireless Localization," *IEEE Journal on Selected Areas in Communications* 41, no. 12 (2023): 3831–3847.

7. S. Gupta, F. Alharbi, R. Alshahrani, et al., "Secure and Lightweight Authentication Protocol for Privacy Preserving Communications in Smart City Applications," *Sustainability* 15, no. 6 (2023): 5346.

8. R. R. Irshad, S. Hussain, I. Hussain, et al., "An Intelligent Buffalo-Based Secure Edge-Enabled Computing Platform for Heterogeneous IoT Network in Smart Cities," *IEEE Access* 11 (2023): 69282–69294.

9. V. O. Nyangaresi, "Provably Secure Authentication Protocol for Traffic Exchanges in Unmanned Aerial Vehicles," *High-Confidence Computing* 3, no. 4 (2023): 100154.

10. M. O. Ahmad, G. Tripathi, F. Siddiqui, et al., "BAuth-ZKP—A Blockchain-Based Multi-Factor Authentication Mechanism for Securing Smart Cities," *Sensors* 23, no. 5 (2023): 2757.

11. X. Hou, L. Zhang, Y. Su, et al., "A Space Crawling Robotic Bio-Paw (SCRBP) Enabled by Triboelectric Sensors for Surface Identification," *Nano Energy* 105 (2023): 108013.

12. W. Zou, Y. Sun, Y. Zhou, et al., "Limited Sensing and Deep Data Mining: A New Exploration of Developing City-Wide Parking Guidance Systems," *IEEE Intelligent Transportation Systems Magazine* 14, no. 1 (2020): 198–215.

13. H. Y. Chien, Y. J. Chen, G. H. Qiu, et al., "A MQTT-API-Compatible IoT Security-Enhanced Platform," *International Journal of Sensor Networks* 32, no. 1 (2020): 54–68.

14. N. Nurelmadina, M. K. Hasan, I. Memon, et al., "A Systematic Review on Cognitive Radio in Low Power Wide Area Network for Industrial IoT Applications," *Sustainability* 13, no. 1 (2021): 338.

15. V. O. Nyangaresi, "Privacy Preserving Three-Factor Authentication Protocol for Secure Message Forwarding in Wireless Body Area Networks," *Ad Hoc Networks* 142 (2023): 103117.

16. C. Liu, T. Wu, Z. Li, T. Ma, and J. Huang, "Robust Online Tensor Completion for IoT Streaming Data Recovery," *IEEE Transactions on Neural Networks and Learning Systems* 34, no. 12 (2022): 10178–10192.

17. J. Cha, S. K. Singh, T. W. Kim, and J. H. Park, "Blockchain-Empowered Cloud Architecture Based on Secret Sharing for Smart City," *Journal of Information Security and Applications* 57 (2021): 102686.

18. M. Adil and M. K. Khan, "Emerging IoT Applications in Sustainable Smart Cities for Covid-19: Network Security and Data Preservation Challenges With Future Directions," *Sustainable Cities and Society* 75 (2021): 103311.

19. P. Chen, H. Liu, R. Xin, et al., "Effectively Detecting Operational Anomalies in Large-Scale IoT Data Infrastructures by Using a GAN-Based Predictive Model," *Computer Journal* 65, no. 11 (2022): 2909–2925.

20. H. Li, Q. Huang, J. Huang, and W. Susilo, "Public-Key Authenticated Encryption With Keyword Search Supporting Constant Trapdoor Generation and Fast Search," *IEEE Transactions on Information Forensics and Security* 18 (2022): 396–410.

21. R. Guo, H. Liu, and D. Liu, "When Deep Learning-Based Soft Sensors Encounter Reliability Challenges: A Practical Knowledge-Guided Adversarial Attack and Its Defense," *IEEE Transactions on Industrial Informatics* 20, no. 2 (2023): 1–13.

22. H. Jiang, M. Wang, P. Zhao, Z. Xiao, and S. Dustdar, "A Utility-Aware General Framework With Quantifiable Privacy Preservation for Destination Prediction in LBSs," *IEEE/ACM Transactions on Networking* 29, no. 5 (2021): 2228–2241.

23. J. Ma and J. Hu, "Safe Consensus Control of Cooperative-Competitive Multi-Agent Systems via Differential Privacy," *Kybernetika* 58, no. 3 (2022): 426–439.

24. X. Li, J. Peng, M. S. Obaidat, F. Wu, M. K. Khan, and C. Chen, "A Secure Three-Factor User Authentication Protocol With Forward Secrecy for Wireless Medical Sensor Network Systems," *IEEE Systems Journal* 14, no. 1 (2019): 39–50.

25. L. Wu, J. Wang, K. K. R. Choo, and D. He, "Secure Key Agreement and Key Protection for Mobile Device User Authentication," *IEEE Transactions on Information Forensics and Security* 14, no. 2 (2018): 319–330.

26. V. O. Nyangaresi, "Lightweight Anonymous Authentication Protocol for Resource-Constrained Smart Home Devices Based on Elliptic Curve Cryptography," *Journal of Systems Architecture* 133 (2022): 102763.

27. Z. Liu, P. Qian, J. Yang, et al., "Rethinking Smart Contract Fuzzing: Fuzzing With Invocation Ordering and Important Branch Revisiting," *IEEE Transactions on Information Forensics and Security* 18 (2023): 1237–1251.

28. M. Dener and A. Orman, "BBAP-WSN: A New Blockchain-Based Authentication Protocol for Wireless Sensor Networks," *Applied Sciences* 13, no. 3 (2023): 1526.

29. A. El Bekkali, M. Essaaidi, and M. Boulmalf, "A Blockchain-Based Architecture and Framework for Cybersecure Smart Cities," *IEEE Access* 11 (2023): 76359–76370.

30. S. Awan, N. Javaid, S. Ullah, A. U. Khan, A. M. Qamar, and J. G. Choi, "Blockchain Based Secure Routing and Trust Management in Wireless Sensor Networks," *Sensors* 22, no. 2 (2022): 411.

31. V. O. Nyangaresi, "A Formally Validated Authentication Algorithm for Secure Message Forwarding in Smart Home Networks," *SN Computer Science* 3, no. 5 (2022): 364.

32. M. Nikooghadam, H. Amintoosi, S. H. Islam, and M. F. Moghadam, "A Provably Secure and Lightweight Authentication Scheme for Internet of Drones for Smart City Surveillance," *Journal of Systems Architecture* 115 (2021): 101955.

33. M. Ghahramani, R. Javidan, and M. Shojafar, "A Secure Biometric-Based Authentication Protocol for Global Mobility Networks in Smart Cities," *Journal of Supercomputing* 76 (2020): 8729–8755.

34. B. Bera, A. K. Das, W. Balzano, and C. M. Medaglia, "On the Design of Biometric-Based User Authentication Protocol in Smart City Environment," *Pattern Recognition Letters* 138 (2020): 439–446.

35. D. Maram, H. Malvai, F. Zhang, et al., "Candid: Can-Do Decentralized Identity With Legacy Compatibility, Sybil-Resistance, and Accountability," in *2021 IEEE Symposium on Security and Privacy (SP)* (San Francisco, CA: IEEE, 2021), 1348–1366.

36. J. Lee, J. Oh, and Y. Park, "A Secure and Anonymous Authentication Protocol Based on Three-Factor Wireless Medical Sensor Networks," *Electronics* 12, no. 6 (2023): 1368.

37. P. Vijayakumar, M. S. Obaidat, M. Azees, S. H. Islam, and N. Kumar, "Efficient and Secure Anonymous Authentication With Location Privacy for IoT-Based WBANs," *IEEE Transactions on Industrial Informatics* 16, no. 4 (2019): 2603–2611.

38. Q. Xie, K. Li, X. Tan, L. Han, W. Tang, and B. Hu, "A Secure and Privacy-Preserving Authentication Protocol for Wireless Sensor Networks in Smart City," *EURASIP Journal on Wireless Communications and Networking* 2021, no. 1 (2021): 1–17.

39. W. Yuanbing, L. Wanrong, and L. Bin, "An Improved Authentication Protocol for Smart Healthcare System Using Wireless Medical Sensor Network," *IEEE Access* 9 (2021): 105101–105117.

40. X. Xia, S. Ji, P. Vijayakumar, J. Shen, and J. J. Rodrigues, "An Efficient Anonymous Authentication and Key Agreement Scheme With Privacy-Preserving for Smart Cities," *International Journal of Distributed Sensor Networks* 17, no. 6 (2021): 1–13.

41. Y. Lu, G. Xu, L. Li, and Y. Yang, "Anonymous Three-Factor Authenticated Key Agreement for Wireless Sensor Networks," *Wireless Networks* 25 (2019): 1461–1475.

42. J. Mo and H. Chen, "A Lightweight Secure User Authentication and Key Agreement Protocol for Wireless Sensor Networks," *Security and Communication Networks* 2019 (2019): 1–17.

43. S. K. Singh, Y. S. Jeong, and J. H. Park, "A Deep Learning-Based IoT-Oriented Infrastructure for Secure Smart City," *Sustainable Cities and Society* 60 (2020): 102252.

44. A. O. Khadidos, S. Shitharth, H. Manoharan, A. Yafoz, A. O. Khadidos, and K. H. Alyoubi, "An Intelligent Security Framework Based on Collaborative Mutual Authentication Model for Smart City Networks," *IEEE Access* 10 (2022): 85289–85304.

45. A. Altaf, H. Abbas, F. Iqbal, M. M. Z. M. Khan, A. Rauf, and T. Kanwal, "Mitigating Service-Oriented Attacks Using Context-Based Trust for Smart Cities in IoT Networks," *Journal of Systems Architecture* 115 (2021): 102028.

46. A. A. Abd El-Latif, B. Abd-El-Atty, I. Mehmood, K. Muhammad, S. E. Venegas-Andraca, and J. Peng, "Quantum-Inspired Blockchain-Based Cybersecurity: Securing Smart Edge Utilities in IoT-Based Smart Cities," *Information Processing & Management* 58, no. 4 (2021): 102549.

47. D. Wang, B. Bai, K. Lei, W. Zhao, Y. Yang, and Z. Han, "Enhancing Information Security via Physical Layer Approaches in Heterogeneous IoT With Multiple Access Mobile Edge Computing in Smart City," *IEEE Access* 7 (2019): 54508–54521.

48. D. Dharminder, D. Mishra, and X. Li, "Construction of RSA-Based Authentication Scheme in Authorized Access to Healthcare Services: Authorized Access to Healthcare Services," *Journal of Medical Systems* 44, no. 1 (2020): 6.

49. D. Chaudhary, T. Soni, S. Singh, and S. M. C. Gupta, "A Construction of Secure and Efficient Authenticated Key Exchange Protocol for Deploying Internet of Drones in Smart City," in *International Conference on Artificial Intelligence of Things* (Cham, Switzerland: Springer Nature, 2023), 136–150.

50. D. Chaudhary, T. Soni, K. L. Vasudev, and K. Saleem, "A Modified Lightweight Authenticated Key Agreement Protocol for Internet of Drones," *Internet of Things* 21 (2023): 100669.

51. H. Sikarwar and D. Das, "A Novel Mac-Based Authentication Scheme (NoMAS) for Internet of Vehicles (IoV)," *IEEE Transactions on Intelligent Transportation Systems* 24, no. 5 (2023): 4904–4916.

52. B. Bera, A. K. Das, S. Garg, M. J. Piran, and M. S. Hossain, "Access Control Protocol for Battlefield Surveillance in Drone Environment," *IEEE Internet of Things Journal* 9, no. 4 (2022): 2708–2721.