A. Irshad, G. A. Mallah, M. Bilal, S. A. Chaudhry, M. Shafiq and H. Song, "SUSIC: A Secure User Access Control mechanism for SDN-enabled IIoT and Cyber Physical Systems," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2023.3268474.

https://doi.org/10.1109/JIOT.2023.3268474

SUSIC: A Secure User Access Control mechanism for SDN-enabled IIoT and Cyber Physical Systems

Azeem Irshad, Ghulam Ali Mallah, Muhammad Bilal, *Senior Member, IEEE,* Shehzad Ashraf Chaudhry, Muhammad Shafiq, Houbing Song, *Fellow, IEEE*

***Abstract*—The integration of thriving Information and Communications Technology (ICT) and Cyber Physical Systems (CPS) has spawned several innovative applications such as remote healthcare, smart and intelligent transportation, smart logistics, smart grids, public safety etc. An emerging Software Defined Networks (SDN) technology further enabled to optimize the communication among industrial IoT (IIoT) and CPS entities. Nonetheless, the communication on public channel among different IIoT entities in an SDN-enabled environment may be exposed to various security threats due to wireless and insecure communication channels. To counter these security challenges in the way of wider CPS or IIoT adoption, we propose a novel three-factor authenticated key exchange mechanism (SUSIC) for SDN-enabled IIoT ecosystem. The SUSIC enables a registered user to access real-time data from physical IIoT environment directly after having mutual authentication performed through SDN-enabled controller node. The scheme is proved to be secure under rigorous formal and informal security analysis. Moreover, the simulation results and performance evaluation signifies towards achieving better trade-off between security functionalities and computational overheads comparatively.**

***Index Terms*—SDN, IIoT, CPS, Industry 4.0, Authenticated key agreement**

I. Introduction

THE IIoT/CPS system facilitates many application domains with the integration of industrial control system and physical network system [1]. Moreover, this system offers a plethora of sustainable up-to-date services by embedding actuators and sensors in many prospective industrial applications such as smart cities and transportation, smart grid, smart healthcare, smart manufacturing, and other industrial automated systems [2] etc. The rapid increase in Industry 4.0 oriented IIoT applications in the last few years emphasizes the need of robust security solutions to safeguard the IIoT-based environment from known attacks. Software-Defined Networking (SDN), a novel architecture, may extensively improve the resilience of IIoT/CPS in terms of security [3], [4]. Besides, SDN offers Quality of Service (QoS), cost-effectiveness, adaptability and management capability to the physical network, enabling smooth functioning of dynamic and high-bandwidth applications. The SDN framework decouples network control and packet forwarding tasks, facilitating directly programmable network control, and abstraction of applications and services from the physical infrastructure. The IIoT/CPS environment comprises a lot of wireless sensors and actuators with limited power resources. Besides QoS features including resource allocation and reliability, the sharp increase in the number of CPS devices in technology landscape further emphasizes the need of stringent security practices for smooth performance of the CPS system by overcoming the known attacks.

A. Motivation

The conventional security practices might not work fully to protect the IIoT-based ecosystem. One of the major challenges is to realize the notion of secure communication by establishing a mutually agreed session key among registered users and heterogeneous CPS devices from different vendors. It is also desirable that the IIoT devices may scale up adequately in the physical CPS network irrespective of the security compliance or enhancement. The attacker in IIoT/CPS-based network may disrupt the physical infrastructure through injection of false data, or may initiate multiple attacks including forgery attacks [5], man-in-the-middle attacks, ephemeral secrets leakage attacks [6], physical device capture attacks [7], traceability and anonymity failure attacks [8]. Thus the current study seeks to provide an efficient and secure authenticated key agreement leading to agreed session key for possibly secure communication among the legal users and smart devices in SDN-enabled IIoT network.

B. Software Defined Networking

The SDN, a new networking paradigm, provides a way to combine software based systems and physical network hardware to enhance networking capabilities [9]. Unlike conventional networks, the SDN technology brings flexibility and openness into the network by decoupling the control flow and data forwarding in the system. The part of SDN managing the control flow and maintaining the

The work of Shehzad Ashraf Chaudhry was supported by the Abu Dhabi University's Office of Research and Sponsored Programs under Grant 19300810.

A. Irshad is with Department of Computer science and Software Engineering, International Islamic University, Islamabad, Pakistan, (irshadazeem2@gmail.com).

G.A. Mallah is with Department of Computer Science, Shah Abdul Latif University, Khairpur, Sindh 66020, Pakistan, (ghulam.ali@salu.edu.pk).

M. Bilal is with Department of Computer Engineering, Hankuk University of Foreign Studies, Yongin-si, Gyeonggido, 17035, Korea, (m.bilal@ieee.org).

S.A. Chaudhry is with department of Computer Science and Information Technology, Abu Dhabi University, Abu Dhabi, United Arab Emirates, (ashraf.shehzad.ch@gmail.com).

M. Shafiq is with Department of Information and Communication Engineering, Yeungnam University, Gyeongsan-si, Gyeongsangbuk-do Daehakro, Republic of Korea, (shafiq@ynu.ac.kr).

H. Song is with the Department of Information Systems, University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250 USA, (h.song@ieee.org).

topology of network is termed as the ”Control plane”, while the portion responsible for forwarding the data packets based on the routes directed through the control plane, is termed as ”data plane”. The original design of SDN did not consider the security concerns that left the system with few potential vulnerabilities. One of most critical security limitation is that it lacks a proper authentication protocol mechanism between controller entity and network applications. Although SDN defines few sets of protocols to program the control plane and switching configurations. OpenFlow was the first standard defined by Open Networking Foundation (ONF) to implement SDN, which defines interface between OpenFlow controller as well as OpenFlow switch [10]. Later on many standards including ONOS, CORD etc. were presented for SDN configuration. The Google, Cisco and IBM have initiated to employ SDN for configuring their data centres. Although many standards have been defined, yet the current SDN protocols require concerted efforts to secure its services.

C. Network Model

The network model of the proposed scheme (SUSIC) is demonstrated in Fig. 1. This model illustrates the SDN-enabled IIoT/CPS environment such as smart health, smart logistics and manufacturing, smart transportation, etc. An industrialized CPS having n_d IoT oriented smart devices, i.e. $\{SD_k \mid 1 \leq k \leq n_d\}$ is deployed with many switches in data plane as a part of SDN network. The switches responsible for forwarding data packets in the data plane, are linked with n_d SDN-enabled controller nodes, say $\{CN_j \mid 1 \leq j \leq n_d\}$ in control plane. In SDN-enabled contributed model, the control plane handles the authentication requests from application layer to verify registered users U_i, this could facilitate in maintaining secure sessions among subscribers and particular smart devices. An additional charge of authenticating registered users for the controller node might overburden the CN_j entity. We can witness several state-of-the-art solutions such as OpenDaylight, HyperFlow and Onix [11] to realize a distributed control plane with the help of logically centralized, but physically isolated controller nodes. The interface connecting the control plane and higher layer entities including users, also termed as ”Northbound” interface, while the interface linking the data plane and lower level entities such as smart devices or access points, is termed as ”Southbound” interface. All entities such as CN_j, SD_k and U_i seek to register from trusted registration centre (RC) by issuing private secrets, public keys and certificates, while the U_i and SD_k, later on, may establish an agreed session key on pubic channel with the help of trusted CN_j.

D. Threat Model

The network entities such as smart devices and users employ insecure public channels to communicate on the network. An adversary may take opportunity to compromise the insecure data being exchanged among the participants. The proposed scheme SUSIC considers a universally accepted ”Dolev-Yao (DY) threat model” [12] to assume the capabilities of an adversary $\mathcal{A}$. According to DY model, the $\mathcal{A}$ may intercept, modify, delete or inject faulty information on the channel during communication. In addition, SUSIC employs another widely known *defacto* threat model, i.e. Canetti and Krawczyk (CK)-adversary model [13]. According to CK-adversary model, the attacker may compromise the session short term secrets, session keys, session states after hijacking the sessions. This model emphasises the use of long term secrets along with short term secrets in the construction of session keys to thwart ephemeral secret leakage (ESL) threats. Besides, $\mathcal{A}$ may physically compromise the IoT based smart and mobile devices of users. Consequently, $\mathcal{A}$ may recover the stored secret credential using power differentiation analysis attack [14]–[16]. The registration centre (RC) is assumed to be fully trusted authority in the IIoT/CPS environment.

Table I: Notations guide

Notations	Narrative
RC	Registration Centre
CN_j	j^{th} Controller node
SD_k	k^{th} IoT smart device in physical network
U_i, MD_{U_i}	i^{th} user having mobile device MD_{U_i}
ID_{CN_j}, SID_{CN_j}	CN_j's legal identity and pseudo-identity
ID_{SD_k}, SID_{SD_k}	SD_k's legal identity and registered identity
ID_{U_i}, SID_{U_i}	U_i's Real identity
PID_{U_i}, TID_{U_i}	U_i's Pseudo-identity and temporal identity
PW_{U_i}, BM_{U_i}	U_i's password and biometrics
pr_{CN_j}/Pub_{CN_j}	Private/public key pair of CN_j
pr_{SD_k}/Pub_{SD_k}	Private/public key pair of SD_k
K_{SD_k,CN_j}	Shared secret between CN_j and SD_k
$Cert_{U_i}, Cert_{SD_k}, Cert_{CN_j}$	Certificates of U_i, SD_k, CN_j respectively
TS_1, TS_2, TS_3	Fresh timestamps of U_i, CN_j, SD_k respectively
RTS_{SD_k}	Registration timestamp of SD_k selected by RC
$Gen(.)/Rep(.)$	Generation/ Reproduction functions in fuzzy extractor
$a \cdot P$	Dot represent ECC-based scalar point multiplication
$\sigma_{U_i}, \beta_{U_i}$	Biometric key and public biometric factor for U_i

E. Contribution

The contribution of the proposed scheme is listed below:

1) We demonstrate a new three factor authentication scheme SUSIC for SDN-enabled IIoT/CPS network which permits an authorized user access real-time data directly from SDN-controlled IoT-based smart devices.
2) The SUSIC is compared against other contemporary studies, and is found to be secure enough against all known attacks with a particular focus on anonymity and untraceability.
3) We have verified the security features of SUSIC using formal security analysis, i.e. Real or Random (RoR) model.
4) The SUSIC supports better security properties notwithstanding the fact, the computational and communication overhead may exceed for a few comparative schemes due to employing asymmetric crypto-primitives. Moreover, various experiments on different crypto-primitives have been carried out to measure the computational costs employing MIRACL library [17].

F. Paper layout

The scheme is organized as mentioned below: Section II describes the related work for CPS security. Section III

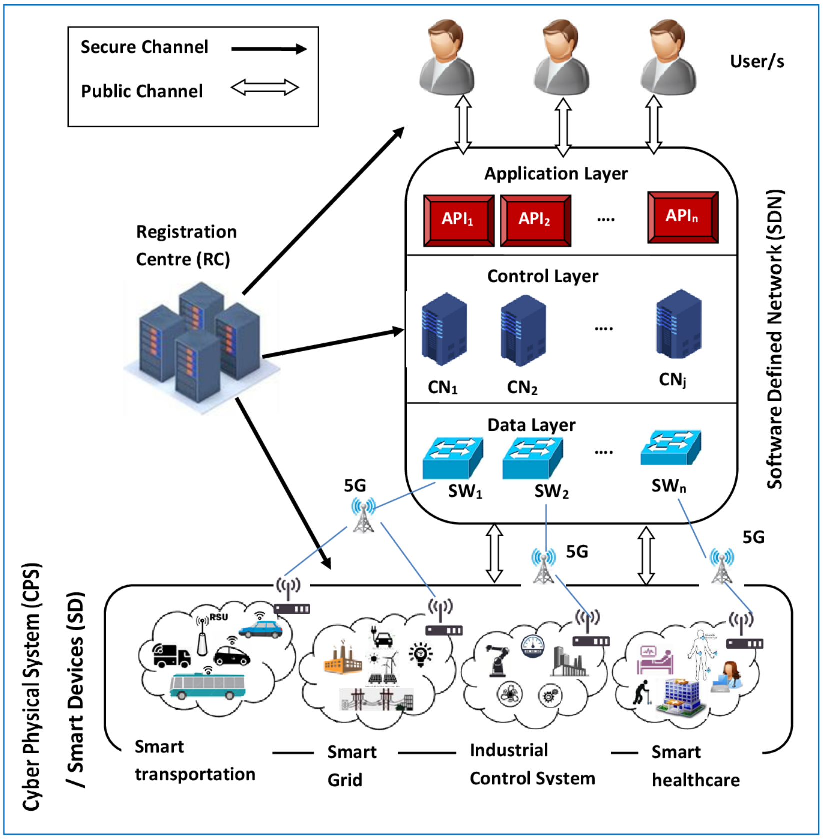


Figure 1: System model depicting SDN-based IIoT/CPS

demonstrates the proposed scheme SUSIC. The security analysis based on formal and informal analysis is described in section IV. Section V evaluates the performance of related schemes. The last section presents the concluding summary.

II. Related work

In this section we bring into focus few significant research studies based on security of industrialized CPS environment. In 2015, NIST [18] elaborated on the security of industrialized control systems. Later, in 2017, the Molina et al. [19] presented a thorough survey on SDN-oriented CPS as well as addressing the cyber security issues of CPS such as physical and logical access to smart devices with the help of access control techniques directed from controller nodes. Taylor et al. [20] and Chong et al. [21] described privacy and security of CPS besides addressing other security challenges as faced by IoT devices in CPS networks. Later, Chen et al. [22] suggested a user authentication protocol for IoT based environment. Although the protocol is efficient in terms of computational and communication delay, it is vulnerable to privileged insider threat as well as lack untraceability feature. In 2020, Bilal et al. [23] demonstrated an authentication protocol for heterogeneous CPS environment in information centric networking. Nonetheless, their scheme is susceptible to ephemeral secret leakage attack under the assumption of CK-attack model, and also lack the dynamic addition of smart IoT devices into the system. In 2019, the Chen et al. [24] designed an authenticated key agreement for smart grid network employing bilinear pairing operations. Despite the costly pairing operations, the scheme is prone to replay and forgery attacks. Bilal and Kang [25] presented another user authenticated key agreement method for group communication in distributed IoT system. The scheme is lightweight from computational and communication perspective, yet it does not support mutual authentication between subscriber and IoT-based smart devices in the network. Renuka et al. [26] introduced a password-based authenticated key exchange method for Machine-to-Machine (M2M)-based CPS systems. This scheme used symmetric encryption for authenticating either user to smart device, or device to device authentication. The scheme [8] presented authentication protocol for 5G-enabled SDN-based CPS, however the scheme does not support mutual authentication since the new timestamp TID^{*}_{Uk} is not verified by controlling authority, which leads the scheme towards de-synchronization for further sessions if maneuvered by the attacker.

More recently, a blockchain-oriented decentralized authentication protocol B-DAC by Duy et al. [27] was introduced for the Northbound interface for managing the critical sources. Next, Alzahrani and Chaudhry [28] presented an identity-based encryption technique for SDN-oriented source routing systems. Recently, another blockchain-based authentication scheme by Vishwakarma et al. [29] was presented for internet of vehicles (IoV). However, the above schemes were costly for employing computation-intensive operations.

In summary, despite there are numerous IoT-based protocols existing in the literature, most of these are either susceptible to security limitations or impractical for practical deployments. In this study, we demonstrate a novel user authenticated key agreement protocol for industrial CPS environment that achieves anonymity and untraceability for user, IoT devices as well as controller node. Moreover, the proposed scheme bears low computational and communication cost, and may successfully thwart the malicious attempts by any CK-model compliant adversary. The proposed scheme, as per the elicited findings, seems to be viable for practical real-time industrial applications.

III. PRELIMINARIES

In this section we illustrate the mathematical preliminaries related to fuzzy extractor and random oracle model.

A. Fuzzy extractor-based Biometric verification

The fuzzy extractor (FE) is an algorithm that produces the same string of output, even if the biometric impression input varies from the pre-recorded biometric impression sample within the permissible threshold of error tolerance [30]. The FE employs two probabilistic and deterministic functions: 1) Generation $Gen(.)$ and Reproduction $Rep(.)$. The $Gen(.)$ takes the biometric input B_{in} and generates a corresponding binary output $B_{op} \in \{0,1\}^l$ using a helper output string $H_{op} \in \{0,1\}^*$. The B_{op} is kept secret and the H_{op} is only stored without keeping secret. To extract B_{op}, B_{in} is merged with H_{op} in $Rep(.)$. The correctness of the FE may be validated using $d_f(B_{in}, B^*_{in}) \leq t$ and $Gen(B_{in}) \rightarrow (B_{in}, H_{in})$. As a result, we get $Rep(B_{in}, H_{op}) \rightarrow (B_{op})$. Whereas d_f indicates distance function, and t be the error threshold.

B. RoR model

The contributed model is analyzed under Random or Real (RoR) model [3], [31] for proving the semantic security, and demonstrate that SUSIC accomplishes the requisite level of session key (SK) security. Considering the RoR model of the contributed SUSIC, the m^{th} instance of entity ξ may be characterized as ξ_m. The subscriber U_i, controller node CN_j, and smart IoT device SD_k are depicted as the entities ξ_{U_i}, ξ_{CN_j} and ξ_{SD_k} along with their m_1^{th}, m_2^{th}, m_3^{th} instances are characterized as $\xi^{m_1}_{U_i}$, $\xi^{m_2}_{CN_j}$ and $\xi^{m_3}_{SD_k}$ respectively. We model the cryptographic one-way hash function h(.) as random oracle H, which may be openly accessed by all entities in RoR model including adversary $\mathcal{A}$. Moreover, this model sets forth the following list of queries that could be initiated by $\mathcal{A}$ during simulation of attack.

- $\mathcal{Q}_{Eav}(\xi^{m_1}_{U_i},\ \xi^{m_2}_{CN_j}, \xi^{m_3}_{SD_k})$: Using this oracle query, an attacker $\mathcal{A}$ may eavesdrop the communication messages exchanged publicly, i.e. M_{sg1}, M_{sg2} and M_{sg3} among entities $\xi^{m_1}_{U_i}, \xi^{m_2}_{CN_j}, \xi^{m_3}_{SD_k}$, respectively, during the establishment of authenticated session key.
- $\mathcal{Q}_{Send}(\xi^m, M_{sg})$: This oracle query permits the adversary to submit message M_{sg} towards ξ^m and receive in response from the same entity. It is also termed as an "active attack".
- $\mathcal{Q}_{Tamper_{MD}}(\xi^m_{U_i})$: Using this query, an adversary may extract all parameters of U_k's registered device MD_{U_i}. This oracle query is analogous to an active attack.
- $\mathcal{Q}_{Reveal}(\xi^m)$: Using this query, the mutually agreed session key $SK_{U_i,SD_k} = (SK_{SD_k,U_i})$ between U_i and SD_k can be exposed to the attacker $\mathcal{A}$.
- $\mathcal{Q}_{Test}(\xi^m)$: The output of $\mathcal{Q}_{Test}$ depends on the outcome of an unbiased coin "ζ" as illustrated below:
  - In case the "$Flip(\zeta) = Head$", it returns the freshly computed session key SK_{U_i,SD_k} between U_i and SD_k.
  - In case "$Flip(\zeta) = Tail$", it will randomly select a number as session key $SK_{U_i,SD_k} \in Z^*_p$, and return SK_{U_i,SD_k} .

C. Design goals

We have the following design goals in this scheme.

- *Confidentiality.* This feature ensures that the participants mutually authenticate one another using the protocol without revealing the secret credentials.
- *Anonymity and untraceability.* The user's real identity should remain confidential and untraceable against the adversary. Untraceability ensures that no two messages from the same source can be either differentiated or linked.
- *Impersonation attack.* $\mathcal{A}$ needs to be debarred from initiating any kind of user or controller node impersonation attack in which the $\mathcal{A}$ attempts to impersonate as a user or controller node.
- *Privileged insider attack.* The protocol should be immune to a insider threats in which $\mathcal{A}$ may access either registration request parameters or the contents stored in smart device, and initiate further attacks.
- *Physical smart device capture attack.* The physical device compromise may help $\mathcal{A}$ to compute the session key established between the other user and victim. However, it should not be able to compute the session keys established among other users.
- *Forward and backward secrecy.* If the private key of a participant is compromised, then previous and future session keys between it and other participants remain secure in the protocol.
- *Ephemeral information leakage threat.* If the short term secrets are known to $\mathcal{A}$, then it may be able to compute the existing session key in the protocol.
- *Man In the Middle attack.* The protocol must eliminate any probability of a malicious intruder attempting to alter the messages during protocol execution with positive verification of messages.

IV. Proposed Scheme

In this section, we demonstrate the three-factor authentication scheme (SUSIC) for SDN-enabled IIoT/CPS network. It supports three-factor authentication which entails all factors such as identity (ID), password, and biometric factors for authenticating the user. The controller node in SDN environment adequately ensures the authenticity of user using FE-enabled three-factor authentication. The SUSIC comprises five important phases, i.e. 1) System initialization 2) Pre-deployment phase 3) User registration phase 4) Login phase 5) Mutual authentication phase. The RC and controller nodes are assumed to be trusted entities in the network. Secondly, all of the involved entities support time-synchronized clocks. The symbols as described in Table I may help the readers to grasp the scheme.

A. System initialization

The system is initialized by the trusted third RC to boostrap important system parameters employing the following steps.

Step 1. The RC selects an adequately large prime integer p, a "non-singular elliptic curve $E_p(a,b) : y^2 = x^3+ax+b(mod\ p)$ over a Galois field $GF(p)$ or a finite field Z_p", given $Z_p = \{0, 1, 2, ..., p-1\}$, assumed a base point P over $E_p(a,b)$ with big order as much as p having infinity $\mathcal{O}$ along with one-way collision resistant hash function $h(.)$.

Step 2. The RC chooses randomly its private key $pr_{RC} \in Z_p^*$ and computes a corresponding public key Pub_{RC} with the use of elliptic curve point (ECC) multiplication, i.e. $Pub_{RC} = pr_{RC}.P$, keeping pr_{RC} as secret while publishing corresponding Pub_{RC} and $h(.)$.

B. Pre-deployment stage

The RC acts as a trusted third party and perform its role in enrolling the controller nodes and smart mobile devices before the deployment.

1) Registering Controller Nodes: The RC performs the following steps to register a Controller Node CN_j.

Step 1. The RC constructs a pseudo-identity $SID_{CN_j} = h(ID_{CN_j}||pr_{RC})$, chooses private key and secret key of CN_j as pr_{CN_j} and $K_{CN_j} \in Z_p^*$, respectively. Then it constructs the related public key $Pub_{CN_j} = pr_{CN_j}.P$. The K_{CN_j} is used to encrypt and decrypt temporary identities for the purpose of synchronization.

Step 2. RC chooses a random integer $n_{CN_j} \in Z_p^*$, and calculates $N_{CN_j} = n_{CN_j}.P$ and constructs a certificate $Cert_{CN_j} = pr_{RC} + h(SID_{CN_j}||Pub_{RC}||Pub_{CN_j}) * n_{CN_j}(mod\ p)$.

Step 3. Next, RC stores $\{SID_{CN_j}, pr_{CN_j}, Pub_{CN_j}, N_{CN_j}, Cert_{CN_j}\}$ and K_{CN_j} safely in the memory of CN_j, while deletes ID_{CN_j}, pr_{CN_j} and $cert_{CN_j}$ from its repository for avoiding privileged insider and stolen-verifier threats.

2) Registering Smart Devices: The RC registers smart devices SD_k with $(1 \leq k \leq m)$ range of devices.

Step 1. RC generates the pseudo-identity $SID_{SD_k} = h(ID_{SD_k}||pr_{RC})$, and constructs a private and public key pair as (pr_{SD_k}, Pub_{SD_k}) after choosing its privacy key $pr_{SD_k} \in Z_p^*$ and corresponding public key $Pub_{SD_k} = pr_{SD_k}.P$. It also generates a secret number randomly $n_{SD_k} \in Z_p^*$ and computes $N_{SD_k} = n_{SD_k}.P$.

Step 2. The RC generates a certificate $Cert_{SD_k} = pr_{RC} + h(SID_{SD_k}||Pub_{RC}||Pub_{SD_k}) * n_{SD_k}(mod\ p)$, where Pub_{CN_j} be the public key of CN_j with which the particular SD_k is associated. Next, it also constructs $K_{SD_k,CN_j} = h(SID_{SD_k}||RID_{CN_j}||n_{SD_k}||n_{CN_j}||RTS_{SD_k})$, where RTS_{SD_k} depicts the registration timestamp of SD_k.

Step 3. Ultimately, the RC preloads $\{SID_{SD_k}, SID_{CN_j}, N_{CN_j}, pr_{SD_k}, Pub_{SD_k}, N_{SD_k}, Cert_{SD_k}, K_{SD_k,CN_j}\}$ in the memory of SD_k, and deletes ID_{SD_k}, RTS_{SD_k}, n_{SD_k} and pr_{SD_k} from its memory for avoiding privileged insider and stolen verifier threats.

Moreover, the RC initializes the smart devices $\{SD_k | 1 \leq k \leq m\}$ with the shared secrets K_{SD_k,CN_j} which are linked to to particular CN_k. In the end, it deletes n_{CN_j} from its memory.

3) User Registration phase: The RC registers U_i intending to join the SDN based industrial CPS network by executing the following steps over secure channel:

Step 1. The U_i chooses its identity as ID_{U_i} and password PW_{U_i}. U_i selects a random number $a_i \in Z_p^*$ and calculates its pseudo-identity $PID_{U_i} = h(ID_{U_i}||a_i)$ for introducing itself to RC. The U_i constructs a private key and public key pair $(pr_{U_i}), Pub_{U_i}$ by selecting $pr_{U_i} \in Z_p^*$, and computing $Pub_{U_i} = pr_{U_i}.P$ to submit registration request $\{PID_{U_i}, Pub_{U_i}\}$ towards RC over secure channel.

Step 2. Upon receiving registration request, the RC computes $SID_{U_i} = h(PID_{U_i}||pr_{RC})$, chooses a random number $n_{U_i} \in Z_p^*$ to compute $N_{U_i} = n_{U_i}.P$. RC selects a randomly generated temporal identity TID_{U_i} for U_i.

Step 3. The RC creates a certificate $Cert_{U_i}$ for U_i, i.e. $Cert_{U_i} = pr_{RC} + h(SID_{U_i}||Pub_{RC}||Pub_{U_i}) * n_{U_i}(mod\ p)$. Then it computes $W_{U_i} = E_{K_{CN_j}}(TID_{U_i})$, submits $\{W_{U_i}, N_{U_i}, Cert_{U_i}\}$ to U_i and deletes n_{U_i} as well as $Cert_{U_i}$ from its memory. Moreover, RC keeps $\{TID_{U_i}, SID_{U_i}, N_{U_i}\}$ securely in the repository of CN_j.

Step 4. The U_i imprints its biometric impression BM_{U_i} and creates biometric secret factor σ_{U_i} and public reproduction factor β_{U_i} by employing the fuzzy extractor-based generation function, i.e. $Gen(BM_{U_i}) = (\sigma_{U_i}, \beta_{U_i})$, and calculates $A_{U_i} = pr_{U_i} \oplus h(\sigma_{U_i}||PID_{U_i}||PW_{U_i})$, $I_{U_i} = a_k \oplus h(PW_{U_i}||\sigma_{U_i}||ID_{U_i})$, $Cert^*_{U_i} = Cert_{U_i} \oplus h(\sigma_{U_i}||PW_{U_i})$ and $Y_{U_i} = h(Cert_{U_i}||SID_{U_i}||PW_{U_i}||N_{U_i}||pr_{U_i})$.

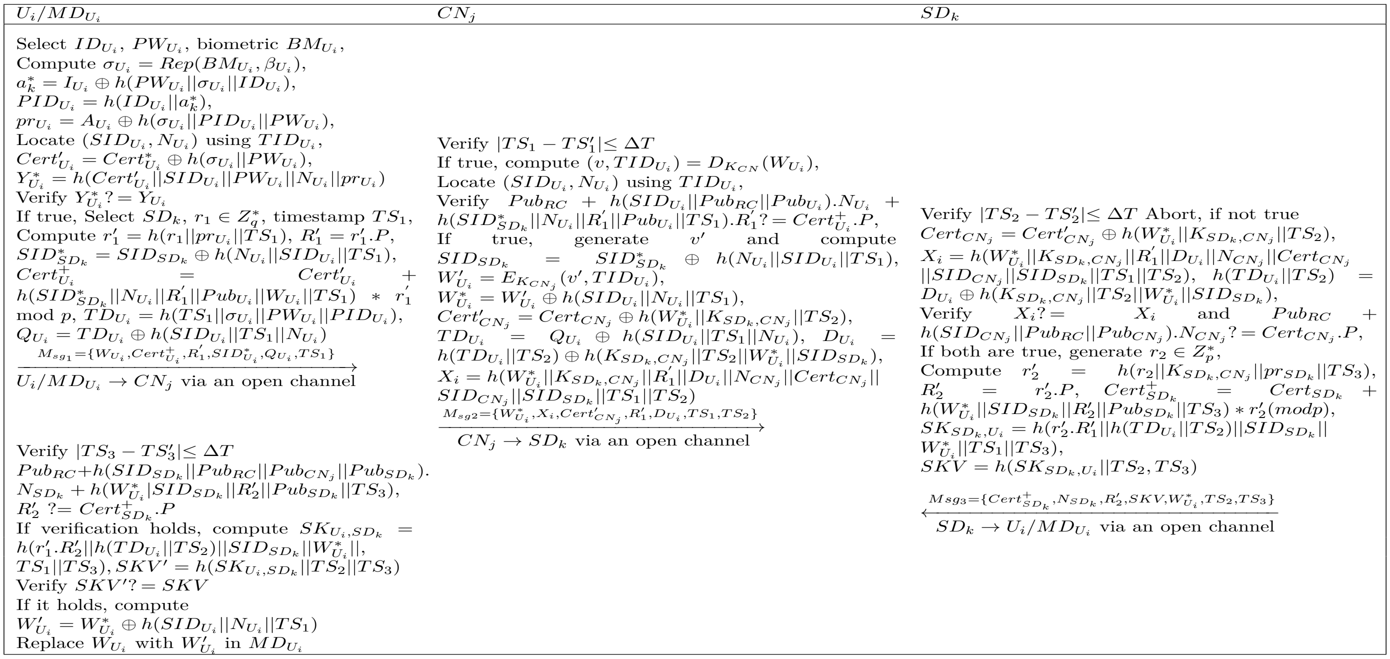
Step 5. Ultimately, U_i erases n_{U_i} and pr_{U_i} from its memory, while removes $Cert^*_{U_i}$ from the MD_{U_i}, and stores $\{Pub_{U_i}, A_{U_i}, I_{U_i}, W_{U_i}, Y_{U_i}, Cert^*_{U_i}, h(.), P, \beta_{U_i}, E_p(a,b), \tau, Gen(.), Rep(.)\}$, where τ represent the error margin threshold for using in fuzzy extractor-based Rep function.

C. Login phase

During login phase, the U_i executes the following steps to login into the industrial CPS environment for interaction with a particular smart device SD_k.

Step 1. Initially, the U_i inputs its identity ID_{U_i}, password PW_{U_i}, and biometric impression BM'_{U_i} on the registered MD_{U_i}. The MD_{U_i} then computes $\sigma_{U_i} = Rep(BM_{U_i}, \beta_{U_i})$ with the provision that the hamming distance of the registered biometric impression BM_{U_i} and the captured current biometric BM'_{U_i} is either less than or equal to predefined error margin threshold τ. Then it further computes $a^*_i = I_{U_i} \oplus h(PW_{U_i}||\sigma_{U_i}||ID_{U_i}), PID_{U_i} = h(ID_{U_i}||a^*_i), pr_{U_i} = A_{U_i} \oplus h(\sigma_{U_i}||PID_{U_i}||PW_{U_i})$ and $Cert'_{U_i} = Cert^*_{U_i} \oplus h(\sigma_{U_i}||PW_{U_i})$.

Step 2. The MD_{U_i} now computes $Y^*_{U_i} =$



U_i/MD_{U_i}

Select ID_{U_i}, PW_{U_i}, biometric BM_{U_i},
Compute $\sigma_{U_i} = Rep(BM_{U_i}, \beta_{U_i})$,
$a^*_k = I_{U_i} \oplus h(PW_{U_i}||\sigma_{U_i}||ID_{U_i})$,
$PID_{U_i} = h(ID_{U_i}||a^*_k)$,
$pr_{U_i} = A_{U_i} \oplus h(\sigma_{U_i}||PID_{U_i}||PW_{U_i})$,
Locate (SID_{U_i}, N_{U_i}) using TID_{U_i},
$Cert'_{U_i} = Cert^*_{U_i} \oplus h(\sigma_{U_i}||PW_{U_i})$,
$Y^*_{U_i} = h(Cert'_{U_i}||SID_{U_i}||PW_{U_i}||N_{U_i}||pr_{U_i})$
Verify $Y^*_{U_i} ?= Y_{U_i}$
If true, Select SD_k, $r_1 \in Z^*_q$, timestamp TS_1,
Compute $r'_1 = h(r_1||pr_{U_i}||TS_1)$, $R'_1 = r'_1.P$,
$SID^*_{SD_k} = SID_{SD_k} \oplus h(N_{U_i}||SID_{U_i}||TS_1)$,
$Cert^+_{U_i} = Cert'_{U_i} + h(SID^*_{SD_k}||N_{U_i}||R'_1||Pub_{U_i}||W_{U_i}||TS_1) * r'_1$ mod p, $TD_{U_i} = h(TS_1||\sigma_{U_i}||PW_{U_i}||PID_{U_i})$,
$Q_{U_i} = TD_{U_i} \oplus h(SID_{U_i}||TS_1||N_{U_i})$

$$\xrightarrow{M_{sg_1}=\{W_{U_i}, Cert^+_{U_i}, R'_1, SID^*_{U_i}, Q_{U_i}, TS_1\}}$$

$U_i/MD_{U_i} \rightarrow CN_j$ via an open channel

CN_j

Verify $|TS_1 - TS'_1| \leq \Delta T$
If true, compute $(v, TID_{U_i}) = D_{K_{CN}}(W_{U_i})$,
Locate (SID_{U_i}, N_{U_i}) using TID_{U_i},
Verify $Pub_{RC} + h(SID_{U_i}||Pub_{RC}||Pub_{U_i}).N_{U_i} + h(SID^*_{SD_k}||N_{U_i}||R'_1||Pub_{U_i}||TS_1).R'_1 ?= Cert^+_{U_i}.P$,
If true, generate v' and compute $SID_{SD_k} = SID^*_{SD_k} \oplus h(N_{U_i}||SID_{U_i}||TS_1)$,
$W'_{U_i} = E_{K_{CN_j}}(v', TID_{U_i})$,
$W^*_{U_i} = W'_{U_i} \oplus h(SID_{U_i}||N_{U_i}||TS_1)$,
$Cert'_{CN_j} = Cert_{CN_j} \oplus h(W^*_{U_i}||K_{SD_k,CN_j}||TS_2)$,
$TD_{U_i} = Q_{U_i} \oplus h(SID_{U_i}||TS_1||N_{U_i})$, $D_{U_i} = h(TD_{U_i}||TS_2) \oplus h(K_{SD_k,CN_j}||TS_2||W^*_{U_i}||SID_{SD_k})$,
$X_i = h(W^*_{U_i}||K_{SD_k,CN_j}||R'_1||D_{U_i}||N_{CN_j}||Cert_{CN_j}|| SID_{CN_j}||SID_{SD_k}||TS_1||TS_2)$

$$\xrightarrow{M_{sg2}=\{W^*_{U_i}, X_i, Cert'_{CN_j}, R'_1, D_{U_i}, TS_1, TS_2\}}$$

$CN_j \rightarrow SD_k$ via an open channel

SD_k

Verify $|TS_2 - TS'_2| \leq \Delta T$ Abort, if not true
$Cert_{CN_j} = Cert'_{CN_j} \oplus h(W^*_{U_i}||K_{SD_k,CN_j}||TS_2)$,
$X_i = h(W^*_{U_i}||K_{SD_k,CN_j}||R'_1||D_{U_i}||N_{CN_j}||Cert_{CN_j} ||SID_{CN_j}||SID_{SD_k}||TS_1||TS_2)$, $h(TD_{U_i}||TS_2) = D_{U_i} \oplus h(K_{SD_k,CN_j}||TS_2||W^*_{U_i}||SID_{SD_k})$,
Verify $X_i ?= X_i$ and $Pub_{RC} + h(SID_{CN_j}||Pub_{RC}||Pub_{CN_j}).N_{CN_j} ?= Cert_{CN_j}.P$,
If both are true, generate $r_2 \in Z^*_p$,
Compute $r'_2 = h(r_2||K_{SD_k,CN_j}||pr_{SD_k}||TS_3)$, $R'_2 = r'_2.P$, $Cert^+_{SD_k} = Cert_{SD_k} + h(W^*_{U_i}||SID_{SD_k}||R'_2||Pub_{SD_k}||TS_3) * r'_2(modp)$,
$SK_{SD_k,U_i} = h(r'_2.R'_1||h(TD_{U_i}||TS_2)||SID_{SD_k}|| W^*_{U_i}||TS_1||TS_3)$,
$SKV = h(SK_{SD_k,U_i}||TS_2, TS_3)$

$$\xleftarrow{Msg_3=\{Cert^+_{SD_k}, N_{SD_k}, R'_2, SKV, W^*_{U_i}, TS_2, TS_3\}}$$

$SD_k \rightarrow U_i/MD_{U_i}$ via an open channel

Verify $|TS_3 - TS'_3| \leq \Delta T$
$Pub_{RC}+h(SID_{SD_k}||Pub_{RC}||Pub_{CN_j}||Pub_{SD_k})$.
$N_{SD_k} + h(W^*_{U_i}|SID_{SD_k}||R'_2||Pub_{SD_k}||TS_3)$,
R'_2 ?= $Cert^+_{SD_k}.P$
If verification holds, compute $SK_{U_i,SD_k} = h(r'_1.R'_2||h(TD_{U_i}||TS_2)||SID_{SD_k}||W^*_{U_i}||$, $TS_1||TS_3), SKV' = h(SK_{U_i,SD_k}||TS_2||TS_3)$
Verify $SKV'? = SKV$
If it holds, compute
$W'_{U_i} = W^*_{U_i} \oplus h(SID_{U_i}||N_{U_i}||TS_1)$
Replace W_{U_i} with W'_{U_i} in MD_{U_i}

Figure 2: Proposed Scheme

$h(Cert'_{U_i}||SID_{U_i}||PW_{U_i}||N_{U_i}||pr_{U_i})$, and then verifies the equality for $Y^*_{U_i}$?= Y_{U_i} which may hold true or false. If it is false, MD_{U_i} drops the login process and terminates the session. Otherwise, the MD_{U_i} chooses a random secret $r_1 \in Z^*_p$, generates fresh timestamp TS_1, and calculates $r'_1 = h(r_1||pr_{U_i}||TS_1), R'_1 = r'_1.P$, $SID^*_{SD_k} = SID_{SD_k} \oplus h(N_{U_i}||SID_{U_i}||TS_1)$, $Cert^+_{U_i} = Cert'_{U_i} + h(SID^*_{SD_k}||N_{U_i}||R'_1||Pub_{U_i}||TS_1) * r'_1$ $(mod\ p)$, where SID_{SD_k} be the pseudo-identity for that particular device SD_k from which the use U_i attempts to approach the real-time captured data. Next, the U_i computes $TD_{U_i} = h(TS1||\sigma_{U_i}||PW_{U_i}||PID_{U_i}), Q_{U_i} = TD_{U_i} \oplus h(SID_{U_i}||TS_1||N_{U_i})$.

Step 3. Next, the MD_{U_i} submits the authentication request $M_{sg_1} = \{W_{U_i}, Cert^+_{U_i}, R'_1, SID^*_{U_i}, Q_{U_i}, TS_1\}$ to CN_j that would proceed for authenticating a particular associated SD_k as shown in Fig. 2.

D. Mutual Authentication phase

The controller node CN_j after receiving the M_{sg1} authentication request performs the following steps: *Step* 1. The CN_j initially checks the freshness of timestamp TS_1 by verifying $|TS_1 - TS'_1| \leq \Delta T$. If the condition is not true, it aborts the session. Otherwise CN_j computes $(v, TID_{U_i}) = D_{K_{CN}}(W_{U_i})$, and finds (SID_{U_i}, N_{U_i}) in its database using TID_{U_i}. Now it computes and verifies $Pub_{RC} + h(SID_{U_i}||Pub_{RC}||Pub_{U_i}).N_{U_i} + h(SID^*_{SD_k}||N_{U_i}||R'_1||Pub_{U_i}||TS_1).R'_1 ?= Cert^+_{U_i}.P$. If it is true, it generates v' and computes $SID_{SD_k} = SID^*_{SD_k} \oplus h(N_{U_i}||SID_{U_i}||TS_1), W'_{U_i} = E_{K_{CN_j}}(v', TID_{U_i}), W^*_{U_i} = W'_{U_i} \oplus h(SID_{U_i}||N_{U_i}||TS_1), Cert'_{CN_j} = Cert_{CN_j} \oplus h(W^*_{U_i}||K_{SD_k,CN_j}||TS_2), TD_{U_i} = Q_{U_i} \oplus h(SID_{U_i}||TS_1||N_{U_i}), D_{U_i} = h(TD_{U_i}||TS_2) \oplus h(K_{SD_k,CN_j}||TS_2||W^*_{U_i}||SID_{SD_k})$ and $X_i = h(W^*_{U_i}||K_{SD_k,CN_j}||R'_1||D_{U_i}||N_{CN_j}||Cert_{CN_j}$ & $||SID_{CN_j}||SID_{SD_k}||TS_1||TS_2)$, where TS_2 is fresh timestamp. Now it submits the request for key establishment $M_{sg2} = \{W^*_{U_i}, X_i, Cert'_{CN_j}, R'_1, D_{U_i}, TS_1, TS_2\}$ to SD_k. *Step* 2. After receiving M_{sg2} from CN_j, the SD_k checks $|TS_2 - TS'_2| \leq \Delta T$. If it does not hold, it discards the request. Or else, it extracts $SID_{SD_k}, SID_{CN_j}, K_{SD_k,CN_j}$ from its database, and computes $Cert_{CN_j} = Cert'_{CN_j} \oplus h(W^*_{U_i}||K_{SD_k,CN_j}||TS_2), X'_i = h(W^*_{U_i}||K_{SD_k,CN_j}||R'_1||D_{U_i}||N_{CN_j}||Cert_{CN_j}||SID_{CN_j} ||SID_{SD_k}||TS_1||TS_2)$, $h(TD_{U_i}||TS_2) = D_{U_i} \oplus h(K_{SD_k,CN_j}||TS_2||W^*_{U_i}||SID_{SD_k})$. Now it verifies $X'_i?=X_i$ and $Pub_{RC} + h(SID_{CN_j}||Pub_{RC}||Pub_{CN_j}).N_{CN_j}?=Cert_{CN_j}.P$. If both equations are true, it generates a random integer $r_2 \in Z^*_p$ and further computes $r'_2 = h(r_2||K_{SD_k,CN_j}||pr_{SD_k}||TS_3), R'_2 = r'_2.P, Cert^+_{SD_k} = Cert_{SD_k} + h(W^*_{U_i}||SID_{SD_k}||R'_2||Pub_{SD_k}||TS_3) * r'_2(modp), SK_{SD_k,U_i} = h(r'_2.R'_1||h(TD_{U_i}||TS_2)||SID_{SD_k}|| W^*_{U_i}||TS_1|| TS_3), SKV = h(SK_{SD_k,U_i}||TS_2, TS_3)$, where TS_3 is fresh timestamp. Now it sends the message $M_{sg3} = \{Cert^+_{SD_k}, N_{SD_k}, R'_2, SKV, W^*_{U_i}, TS_2, TS_3\}$ to U_i for final verification.

Step 3. Upon receiving M_{sg3} from SD_k, the U_i verifies the timestamp by $|TS_3 - TS'_3| \leq \Delta T$. It aborts the session, if does not hold. Otherwise, MD_{U_i} verifies $Pub_{RC} + h(SID_{SD_k}||Pub_{RC}||Pub_{CN_j}||Pub_{SD_k}).N_{SD_k} + h(W^*_{U_i}|SID_{SD_k}||R'_2||Pub_{SD_k}||TS_3).R'_2 ?= Cert^+_{SD_k}.P$ If verification holds, it further computes the session key $SK_{U_i,SD_k} = h(r'_1.R'_2||h(TD_{U_i}||TS_2)||SID_{SD_k}||W^*_{U_i}||TS_1||TS_3)$ as well as the session key verifier $SKV' = h(SK_{U_i,SD_k}||TS_2||TS_3)$. Next, it verifies the equality $SKV'?= SKV$ to validate the session key. If this holds true, it further calculates $W'_{U_i} = W^*_{U_i} \oplus h(SID_{U_i}||N_{U_i}||TS_1)$, and replaces W_{U_i} with W'_{U_i} in MD_{U_i} in its repository. Now, having a mutually agreed session key, the U_i and SD_k may proceed for further communication.

V. Security Analysis

This section demonstrates the resilience of contributed scheme (SUSIC) using formal and informal security analysis as elaborated in the following:

A. Formal Security Analysis

We describe RoR model in context of proposed SUSIC, and then evaluate the SK-security of proposed model in the Theorem 1. The scheme [35] depicts that the verifiers such as user passwords are not distributed on uniform basis, and are under constraint with a fairly limited set in the domain of permitted passwords. We apply the Zipf's law for proving the session key security of SUSIC. The proof for SK-security of SUSIC is provided in Theorem 1.

Theorem 1: The $Adv_{\mathcal{A}}^{AKA-SUSIC}(t)$ is assumed to be the advantage for polynomial time adversary $\mathcal{A}$ to break the SK-security of proposed authenticated key agreement scheme (AKA-SUSIC). Let q_{hs}, q_{Ev}, $|H|$ and $Adv_{\mathcal{A}}^{AKA-SUSIC}(t)$ be the number of hash function queries, number of eavesdrop queries, range space for hash-digest function h(.), and advantage of breaking of elliptic curve Diphie-Hellman Problem (ECDHP) in polynomial time t, respectively. If l_σ be the length of biometric secret key σ_{U_i} for user, C' and s' represent the Zipf's oriented factors, we get the following equation:

$$Adv_{\mathcal{A}}^{AKA-SUSIC}(t) \leq \frac{q_{hs}^2}{|H|} + 2[max\{C'.q_{Ev}^{s'}, \frac{q_{Ev}}{2^{l_\sigma}}\} + Adv_{\mathcal{A}}^{ECDHP}(t)] \tag{1}$$

Proof: Referring to similar proofs as provided in [36]-[38], we simulate four games Gm_0, Gm_1, Gm_2, Gm_3 associated with an event $Succ_{\mathcal{A}}^{Gm_g}$ as the adversary makes a guess of the outcome of flipped coin ζ regarding Gm_g precisely. In this context, we can describe the advantage of $\mathcal{A}$ for winning the game Gm_g as $Adv_{\mathcal{A}}^{Gm_g} = Pr[Succ_{\mathcal{A}}^{Gm_g}]$. We discuss the details of games Gm_0, Gm_1, Gm_2, Gm_3 as given below.

Game Gm_0: The game Gm_0 models an actual attack as executed by the adversary against the proposed scheme in RoR model. The outcome of the flipped coin ζ is randomly selected in the beginning of Gm_0, thus

$$Adv_{\mathcal{A}}^{AKA-SUSIC}(t) = |2.Adv_{\mathcal{A}}^{Gm_0} - 1| \tag{2}$$

Game Gm_1: The game Gm_0 simulates an "eavesdropping attack" where the adversary attempts to access the publicly available messages on insecure channel, i.e. $M_{sg_1} = \langle W_{U_i}, Cert_{U_i}^+, R_1^{'}, SID_{U_i}^*, Q_{U_i}, TS_1 \rangle$, $M_{sg2} = \langle W_{U_i}^*, X_i, Cert'_{CN_j}, R'_1, D_{U_i}, TS_1, TS_2 \rangle$, $M_{sg3} = \langle Cert_{SD_k}^+, N_{SD_k}, R'_2, SKV, W_{U_i}^*, TS_2, TS_3 \rangle$ as submitted from U_i to CN_j, CN_j to SD_k and SD_k to U_i, respectively during authenticated key exchange by employing oracle query $\mathcal{Q}_{Eav}$. Thereafter, $\mathcal{A}$ launches $\mathcal{Q}_{Reveal}$ and $\mathcal{Q}_{Test}$ queries to test whether the session key SK_{U_i,SD_k} is a valid key or just a random number. As elaborated above, the session key $SK_{SD_k,U_i} = h(r'_2.R'_1||h(TD_{U_i}||TS_2)||SID_{SD_k}||W^*_{U_i}||TS_1||TS_3) = SK_{U_i,SD_k} = SK_{U_i,SD_k} = h(r'_1.R'_2||h(TD_{U_i}||TS_2) ||SID_{SD_k}||W^*_{U_i}||TS_1||TS_3)$ is constructed using the ephemeral secret parameters r_1 and r_2 along with high entropy long terms secret key SID_{U_i} which is unknown to the adversary. Thus, $\mathcal{A}$ may not differentiate a legitimate session SK_{SD_k,U_i} from a random integer. This is because, the Gm_0 and Gm_1 are not distinguishable, thus we have

$$Adv_{\mathcal{A}}^{Gm_1} = Adv_{\mathcal{A}}^{Gm_0} \tag{3}$$

Game Gm_2: The game Gm_2 attempts to simulate an active attack through modeling H oracle. It is evident that few critical parameters such as $\langle SID^*_{U_i}, Cert^+_{U_i}, SID_{SD_k} \rangle, \langle X_i, Cert^{'}_{CN_j,D_{U_i}} \rangle, \langle Cert^+_{SD_k}, SKV \rangle$ as included in the public messages M_{sg1}, M_{sg2} and M_{sg3}, respectively are safe guarded using the collision-resistant cryptographic one-way hash digest function $h(.)$. It is computationally not feasible to extract $W^*_{U_i}, K_{SD_k,CN_j}$ from X_i, and SK_{SD_k,U_i} from SKV owing to one-way property of $h(.)$ function. In addition, the usage of fresh timestamps $[TS_w | 1 \leq w \leq 3]$ and short term ephemeral keys r_1 and r_2 meant for one-time use, the communicative messages M_{sg1}, M_{sg2} and M_{sg3} remain indistinguishable, and in return the collision resistance feature is assured. It is quite obvious that Gm_1 and Gm_2 remain indistinguishable, with the exception that the Gm_2 consists of H-based query simulation. Thus, the understated outcome may be deduced by employing the birthday paradox [18].

$$|Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_2}| \leq \frac{q_{hs}^2}{2|hash|} + Adv_{\mathcal{A}}^{ECDHP} \tag{4}$$

Game Gm_3: In game Gm_3, the attacker attempts to tamper the smart device MD_{U_i} of a particular user U_i by launching oracle queries to $\mathcal{Q}_{TamperMD}$. We assume that $\mathcal{A}$ is in possession for the device MD_{U_i} as well its data contents $\langle SID_{SD_k}, SID_{CN_j}, R_{CN_j}, R_{SD_k}, Cert_{SD_k}, K_{SD_k,CN_j}, (pr_{SD_k}, Pub_{SD_k}) \rangle$. It is computationally not feasible for the attacker to calculate critical factors such as $a_i = I_{U_i} \oplus h(PW_{U_i}||\sigma_{U_i}||ID_{U_i}), PID_{U_i} = h(ID_{U_i}||a_i), pr_{U_i} = A_{U_i} \oplus h(\sigma_{U_i}||PID_{U_i}||PW_{U_i})$ and $Cert'_{U_i} = Cert^*_{U_i} \oplus h(\sigma_{U_i}||PW_{U_i})$ using $\mathcal{Q}_{Send}$ query, irrespective of the knowledge of ID_{U_i}, PW_{U_i} and σ_{U_i}. Furthermore, the probability for guessing biometric key value σ_{U_i} with l_σ bits is calculated as $\frac{1}{2^{l_\sigma}}$ approximately. We also conclude that the games Gm_2 and Gm_3 stand identical, if there is no password or biometric guessing attack. Thus, according to Zipf's law related to passwords, we infer the following equation.

$$|Adv_{\mathcal{A}}^{Gm_2} - Adv_{\mathcal{A}}^{Gm_3}| \leq max\{C^{'}.q_{Ev}^{s^{'}}, \frac{q_{Ev}}{2^{l_\sigma}}\} \tag{5}$$

where the factors $C^{'}$ and $s^{'}$ are called Zipf's parameters [35].

In addition, the advantage of the adversary to guess the outcome after having tossed the coin $\mathcal{C}$, upon execution of the games $[Gm_\phi | 0 \leq \phi \leq 3]$ is $Adv_{\mathcal{A}}^{Gm_3} = \frac{1}{2}$. We obtain the following Eq.(1) and Eq. (2):

$$\begin{aligned}\frac{1}{2}.Adv_{\mathcal{A}}^{AKA-SUSIC}(t) = |Adv_{\mathcal{A}}^{Gm_0} - \frac{1}{2}| = |Adv_{\mathcal{A}}^{Gm_1} - \frac{1}{2}| \\ = |Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_3} - |\end{aligned} \tag{6}$$

In view of Eq.(3)-Eq.(5) upon the application of triangular inequality, we get the following relation.

$$\begin{aligned}&\frac{1}{2}.Adv_{\mathcal{A}}^{AKA-SUSIC}(t) = |Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_3}| \\ &\leq |Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_2}| + |Adv_{\mathcal{A}}^{Gm_2} - Adv_{\mathcal{A}}^{Gm_3}| \\ &\leq \frac{q_{hs}^2}{2|H|} + Adv_{\mathcal{A}}^{ECDHP}(t) + max\{C^{'}.q_{Ev}^{s'}, \frac{q_{Ev}}{2^{l_\sigma}}\}\end{aligned} \tag{7}$$

Ultimately, we obtain the the following Eq. (6) after multiplying with a factor of 2, i.e.

$$\begin{aligned}Adv_{\mathcal{A}}^{AKA-SUSIC}(t) \leq \frac{q_{hs}^2}{|H|} + 2[max\{C^{'}.q_{Ev}^{s'}, \frac{q_{Ev}}{2^{l_\sigma}}\} \\ + Adv_{\mathcal{A}}^{ECDHP}(t)]\end{aligned} \tag{8}$$

B. Informal Security Analysis

This section elaborates informal security analysis on proposed protocol as given below.

1) User impersonation attack: For impersonating as a user, an adversary $\mathcal{A}$ requires to construct a legal authentication request message $M_{sg_1} = \{W_{U_i}, Cert^{+}_{U_i}, R^{'}_1, SID^{*}_{U_i}, Q_{U_i}, TS_1\}$ for submission to CN_j node. An adversary may select a random integer $r_{\mathcal{A}}$ and calculate $R_{\mathcal{A}} = r_{\mathcal{A}}.P$ in addition to selecting a fresh timestamp $TS_{\mathcal{A}}$ during the construction of fake authentication request. However, $\mathcal{A}$ may not generate a legal $Cert^{+}_{U_i} = Cert^{\prime}_{U_i} + h(SID^{*}_{SD_k}||N_{U_i}||R^{'}_1||Pub_{U_i}||W_{U_i}||TS_1) * r^{'}_1(modp)$ since it is ignorant about critical parameters N_{U_i}, SID_{U_i}, $Cert^{\prime}_{U_i}$ and pr_{U_i}. Even if, the $\mathcal{A}$ accesses the contents N_{U_i} and SID_{U_i} physically from U_i's smart device MD_{U_i}, the other parameters $Cert^{'}_{U_i}$ and pr_{U_i} remains masked and protected with PID_{U_i}, PW_{U_i} and σ_{U_i} in MD_{U_i}, the adversary might not compute a legal certificate $Cert^{+}_{U_i}$ and pr_{U_i}. Thus, $\mathcal{A}$ cannot impersonate as legal user in SUSIC.

2) CN_j impersonation attack: An adversary may attempt to forge the controller entity CN_j by calculating a legal message $M_{sg2} = \{W^{*}_{U_i}, X_i, Cert^{\prime}_{CN_j}, R^{\prime}_1, D_{U_i}, TS_1, TS_2\}$. Nonetheless, $\mathcal{A}$ might not do so, since it cannot calculate legal $X_i = h(W^{*}_{U_i}||K_{SD_k,CN_j}||R^{'}_1||D_{U_i}||N_{CN_j}||Cert_{CN_j}$ $||SID_{CN_j}||SID_{SD_k}||TS_1||TS_2)$ without access to the shared secret K_{SD_k,CN_j}. If a smart device gets compromised by the adversary with access to all of its contents including the shared secret K_{SD_k,CN_j} by the adversary, the impact for the compromise of SD_k remains limited to that device only. This is because, the K_{SD_k,CN_j} is unique for each device, thus may not compromise the other devices in the whole CPS system. Moreover, this compromise does not impact the user and its secret credentials remain protected. Hence, the SUSIC is immune to CN_k impersonation attack.

3) SD_k impersonation attack: The adversary may attempt to impersonate as a smart device SD_k in CPS system through fabricating a legitimate message $Msg_3 = \{Cert^{+}_{SD_k}, N_{SD_k}, R^{\prime}_2, SKV, W^{*}_{U_i}, TS_2, TS_3\}$ and submit towards U_i in response to authentication request. However, the adversary might not be successful, since for producing a legal $Cert^{+}_{SD_k}$, it requires knowledge of $Cert_{SD_k} = pr_{RC} + h(SID_{SD_k}||Pub_{RC}||Pub_{SD_k}) * n_{SD_k}(mod\ p)$. Here, $\mathcal{A}$ may also capture the device SD_k and recover the certificate $Cert_{SD_k}$, yet the damaging effect remains local, and compromising of a particular SD_k may not impact the whole CPS system, and neither it exposes an secret credentials of controller entity CN_j, legal user U_i, or other devices SD_k in the system. Thus, SUSIC is resistant to any kind of SD_k impersonation attack

4) U_i's anonymity and untraceability: The SUSIC does not employ the user's identity ID_{U_i} in any communication message on public channel during mutual authentication phase. In case an adversary intercepts the authentication request message $M_{sg_1} = \{W_{U_i}, Cert^{+}_{U_i}, R^{'}_1, SID^{*}_{U_i}, Q_{U_i}, TS_1\}$ over insecure channel, in addition to the compromised U_i's mobile device MD_{U_i} and recovered $\{Pub_{U_i}, A_{U_i}, I_{U_i}, W_{U_i}, Cert^{*}_{U_i}, h(.), P, \beta_{U_i}, E_p(a,b), \tau,$ $Gen(.), Rep(.)\}$ from it. Due to the cryptographic one-way hash function h(.), it is not feasible to recover the identity ID_{U_i} of U_i from $A_{U_i} = pr_{U_i} \oplus h(\sigma_{U_i}||PID_{U_i}||PW_{U_i})$, $I_{U_i} = a_k \oplus h(PW_{U_i}||\sigma_{U_i}||ID_{U_i})$ without knowledge of pr_{U_i}, PW_{U_i}, σ_{U_i} and a_k. Moreover, several authentication request messages originated from the same U_i remains untraceable for the adversary, since W_{U_i} is unique in every new authentication request message. Thus, SUSIC supports anonymity and untracebility for the user.

5) Privileged insider threat: In case the adversary, assuming it a privileged insider referring to RC, reads the U_i's registration request $\{PID_{U_i}, Pub_{U_i}\}$ as submitted to RC over a confidential channel. After the registration procedure, that malicious privileged insider may also get access to the U_i's mobile device MD_{U_i} and its contents. Even then, the attacker may not recover ID_{U_i} or a_k by employing PID_{U_i} due to the property of collision resistance of hash function. In addition, the attacker may not recover crucial secret parameters pr_{U_i}, a_k ad $Cert_{U_i}$ from A_{U_i}, I_{U_i} and $Cert^{*}_{U_i}$ respectively, without access to ID_{U_i}, PW_{U_i} and σ_{U_i}. Thus SUSIC is immune to privileged insider threat.

6) Stolen Mobile Device Threat: Assume that an attacker steals the mobile device MD_{U_i} of any legal user U_i. $\mathcal{A}$ cannot recover the sensitive information a_k, pr_{U_i} and $Cert_{U_i}$ without getting knowledge of U_i's identity ID_{U_i}, password PW_{U_i}, and its biometric secret key σ_{U_i} which may be extracted from U_i's biometric information employing fuzzy extractor mechanisms. The illegitimate modification in N_{U_i} at MD_{U_i} only leads to login failure during login validation procedure. At the same time, illegal update in W_{U_i} parameter can only lead to validation failure that might follow session abort. Thus, SUSIC protocol does not expose any critical information of legal participants in the authentication phase on account of stolen device MD_{U_i}.

7) Compromised Smart Device attack: It is assumed that the attacker gets physical access to SD_k's smart device for malicious objectives and extracts its contents $\{SID_{SD_k}, SID_{CN_j}, N_{CN_j}, pr_{SD_k}, Pub_{SD_k}, N_{SD_k}, Cert_{SD_k}, K_{SD_k,CN_j}\}$ using power analysis differentiation [22]. It is noteworthy that these values are unique for each smart device. In addition, the factors r_2, R_1, W_{U_i}, TS_1 and TS_3 that are employed to calculate the session key SK_{SD_k,U_i} between U_i and SD_k are unique for each session or the smart device. Thus, as a result of a compromised smart device SD_k only session key could be computed between U_i and the compromised device, however it might not aid the adversary to compute the session key as established between U_i and the rest of non-compromised smart devices. Hence, SUSIC is protected from captured smart device threat.

8) Denial-of-Service attack: After receiving the message $M_{sg_1} = \{W_{U_i}, Cert^{+}_{U_i}, R_1^{'}, SID^{*}_{U_i}, Q_{U_i}, TS_1\}$ from U_i, the CN_j checks the freshness of timestamp TS_1 by verifying $|TS_1 - TS_1^{'}| \leq \Delta T$. If it does not match, the CN_j shall not proceed for computing $(v, TID_{U_i}) = D_{K_{CN}}(W_{U_i})$, locating (SID_{U_i}, N_{U_i}) in the repository using TID_{U_i}, and even computing $Cert^{+}_{U_i}.P$ for further proceedings. Thus the proposed scheme can thwart denial of service attack since no adversary can replay the message M_{sg_1} towards CN_j to deplete its resources with fake requests.

Table II: Security Features

	[22]	[32]	[33]	[34]	[24]	[8]	Our
SF_{C1}	✓	✓	✗	✓	✗	✗	✓
SF_{C2}	✗	✗	✗	✓	✗	✓	✓
SF_{C3}	✓	✓	✗	✓	✗	✓	✓
SF_{C4}	✓	✓	✓	-	✓	✓	✓
SF_{C5}	✓	✓	✗	✓	✓	✓	✓
SF_{C6}	✓	✗	✓	✓	✗	✓	✓
SF_{C7}	✗	✗	✓	✓	✓	✓	✓
SF_{C8}	✓	✓	✓	✓	✓	✓	✓
SF_{C9}	✓	✓	✓	✓	✗	✓	✓
SF_{C10}	✓	✓	✗	✓	✗	✓	✓
SF_{C11}	✓	✓	✓	✓	✓	✓	✓
SF_{C12}	✓	✓	✓	✓	✓	✗	✓
SF_{C13}	✓	-	✓	✓	✓	✓	✓
SF_{C14}	✓	✓	✓	✗	✓	✓	✓

Note: SF_{C1}: Supports mutual authentication, SF_{C2}: Supports Anonymity and untraceability, SF_{C3}: $U_i/CN_j/SD_k$ impersonation attack, SF_{C4}: Offline-Password guessing attack, SF_{C5}: Stolen verifier attack, SF_{C6}: Man-in-the-middle attack, SF_{C7}: Ephemeral information leakage attack under CK-model, SF_{C8}: Physical Smart device capture attack, SF_{C9}: Replay attack, SF_{C10}: Privileged insider attack, SF_{C11}: Resist Denial of service attack, SF_{C12}: Resist De-synchronization attack, SF_{C13}: Local modification of password and biometric of user, SF_{C14}: Formal security evaluation; ✓: Resists attack/Supports security functionality, ✗: Do not resist attack or support security functionality.

Table III: Crypto-primitives cost under simulation

CPU	CN/S(msec.)	U/SD (msec.)
T_h	0.026	0.315
$T_{ecm} \approx T_{fe}$	2.173	4.483
T_{eca}	0.089	0.103
T_{sym}	0.055	0.098
T_{bp}	6.251	28.943
T_e	0.042	0.185
T_{IDe}	3.396	18.621
T_{IDd}	11.653	42.905

CPU: Cryptographic Primitives Used; CN/S: Controlling Node/Server; U/SD: User/Smart Device.

9) Password guessing attack: The password PW_{U_i} of user U_i is never communicated in plaintext and remain in masked form in the smart device, and is only used during login validation phase. Thus, online password guessing threat is naturally relaxed in the proposed SUSIC protocol. However, the adversary may attempt to guess the password PW_{U_i} on offline basis by tampering the U_i's mobile device MD_{U_i} through recovering the contents $\{Pub_{U_i}, A_{U_i}, I_{U_i}, W_{U_i}, Y_{U_i}, Cert^{*}_{U_i}, h(.), P, \beta_{U_i}, E_p(a,b), \tau, Gen(.), Rep(.)\}$. However, for successfully guessing PW_{U_i} by employing A_{U_i}, I_{U_i}, $Cert^{*}_{U_i}$ and Y_{U_i}, the adversary must be familiar with either of these tuples $\langle pr_{U_i}, \sigma_{U_i}, PID_{U_i}\rangle$, $\langle a_k, \sigma_{U_i}, ID_{U_i}\rangle$, $\langle Cert_{U_i}, \sigma_{U_i}\rangle$, $\langle Cert_{U_i}, pr_{U_i}\rangle$ which is not feasible. Moreover, extracting pr_{U_i} from public key $Pub_{U_i} = pr_{U_i}.P$ is a hard problem and bounded by the difficulty of Elliptic Curve Discrete Logarithm Problem (ECDLP). Consequently, the adversary may not guess the U_{U_i}'s password PW_{U_i} on offline basis after accessing the mobile device MD_{U_i}.

10) Resists Replay attack: The proposed scheme is resistant to replay attacks as it employs timestamps TS_1, TS_2, TS_3 to construct all communication messages M_{sg1}, M_{sg2}, M_{sg3} respectively. In this manner, the receiver of the message may validate the freshness of received message accurately. Thus, SUSIC is fully resistant to replay attack.

11) Man-in-the-middle attack (MITM): An adversary may attempt to manipulate the authentication request message $M_{sg_1} = \{W_{U_i}, Cert^{+}_{U_i}, R_1^{'}, SID^{*}_{U_i}, Q_{U_i}, TS_1\}$ for initiating a MITM attack between U_i and CN_j [35]. Nonetheless, the attacker might not succeed in its malicious move, since for constructing a legal $Cert^{+}_{U_i}$, it must require the knowledge of pr_{U_i}, r_1, SID_{U_i} and N_{U_i}. A registered but malicious user $U_i^{'}$ acting as an adversary may construct a valid $Cert^{+}_{U_i}{}'$ for itself, however, it cannot construct a valid $Cert^{+}_{U_i}{}''$ for any user $U_i^{'}{}'$ since it must require $\langle SID_{U_i}, N_{U_i}\rangle$ for the corresponding W_{U_i}. Similarly, $\mathcal{A}$ may not be successful in deciphering the intercepted message $M_{sg2} = \{W^{*}_{U_i}, X_i, Cert^{\prime}_{CN_j}, R_1^{\prime}, D_{U_i}, TS_1, TS_2\}$ in the way from CN_j to SD_k, since it must need K_{SD_k,CN_j} as well as pr_{CN_j} to produce a legal $Cert^{'}_{CN_j}$. Likewise, $\mathcal{A}$ cannot manipulate the message $Msg_3 = \{Cert^{+}_{SD_k}, N_{SD_k}, R_2^{\prime}, SKV, W^{*}_{U_i}, TS_2, TS_3\}$ as submitted from SD_k to U_i due to the unknown factor pr_{SD_k}. Hence, our scheme can resist MITM attacks.

12) Ephemeral secret leakage threat: In SUSIC, the protocol generates the session key SK_{SD_k,U_i} on account of ephemeral secrets such as r_1 or r_2 which are freshly generated each time a new session is created. A smart device SD_k in SUSIC calculates the session key as $SK_{SD_k,U_i} = h(r_2^{\prime}.R_1^{\prime}||h(TD_{U_i}||TS_2)||SID_{SD_k}||W^{*}_{U_i}||TS_1|| \quad TS_3) = SK_{U_i,SD_k} = h(r_1^{\prime}.R_2^{\prime}||h(TD_{U_i}||TS_2)||SID_{SD_k}||W^{*}_{U_i}|| TS_1||TS_3)$. We scrutinize the security of session key applying the following two cases:

- *Case* 1. In case the ephemeral secrets r_1 and r_2 are known to the attacker, even then the latter may not calculate the session key K_{SD_k,U_i} without employing the high entropy long term secrets including pr_{U_i}, which are protected under collision resistant feature of one way hash function.
- *Case* 2. In case, the adversary comes to know about long term secret pr_{U_i}, SID_{U_i}, still it would be impossible to construct the session key without knowledge

Table IV: Computational costs

	Smart/Mobile device(msec.)	Controller node(msec.)
[22]	$8T_h + 5T_{ecm} \approx 11.073$	$7T_h + T_{ecm} \approx 2.355$
[32]	$6T_h + 2T_e + T_{sym} \approx 0.295$	$5T_h + 2T_e + T_{IDd} \approx 44.85$
[33]	$8T_h + 9T_{ecm} + T_{fe} \approx 21.938$	$4T_h + 5T_{ecm} \approx 23.675$
[34]	$14T_h+ 2T_b + T_{ecm} \approx 15.039$	$2T_b + 3T_h \approx 58.831$
[24]	$8T_h + 7T_{ecm} + 2T_{eca} + 2T_e + 4T_{bp} \approx 40.685$	–
[8]	$T_{fe} + 24T_h + 9T_{ecm} + 3T_{eca} \approx 22.621$	$9T_h + 3T_{ecm} + 2T_{eca} \approx 16.49$
Our	$T_{fe} + 25T_h + 3T_{eca} + 9T_{ecm} \approx 22.647$	$9T_h + T_{eca} + 2T_{ecm} + 2T_{sym} \approx 12.1$

Table V: Communication Cost Analysis

Scheme.→	[22]	[32]	[33]	[34]	[24]	[8]	Our
Communication cost (bits)	2976	4064	2528	1920	2112	3200	3040

of r_1 and r_2. Thus it is evident from the above cases that the session key can only be produced with the combination of both short term secret as well as long term secret. Leakage of either of the two might not expose the future session keys established between SD_k and U_i. Thus SUSIC is protected against ephemeral secret leakage attack.

C. Security Verification Using ProVerif

We employed ProVerif simulation tool [36] for performing the automated verification of the protocol in order to study its security properties including session key agreement and mutual authentication with the consideration of Canetti-Krawczyk (CK)-based threat Model. Considering the strong security attributes of calculus, it sustains digital signatures, hash digest function, and Public key infrastructure (PKI)-oriented encryption and other strong primitives. To simulate the SASIC system, we modeled three events for the participating entities such as U_i, CN_j and SD_k. In this regard, the events $beginCN_j(bitstring)$ and $endCN_j(bitstring)$ further initialize the corresponding events such as U_i and SD_k through the registration of involved processes. Likewise, the events of the U_i are $beginU_i(bitstring)$ and $endU_i(bitstring)$ to authenticate CN_j. At the same time, the events such as $beginSD_k(bitstring)$ and $endSD_k(bitstring)$ are conjured by SD_k to authenticate U_i. Upon the result computation, it stands to reason that the scheme analyzed that the order of the three pairs of events remains solid. The findings in Fig. 3 depict that contributed model achieves the mutual authenticity through constructing a mutually approved session key among three processes such as U_i, SD_k and CN_j.

VI. Performance evaluation

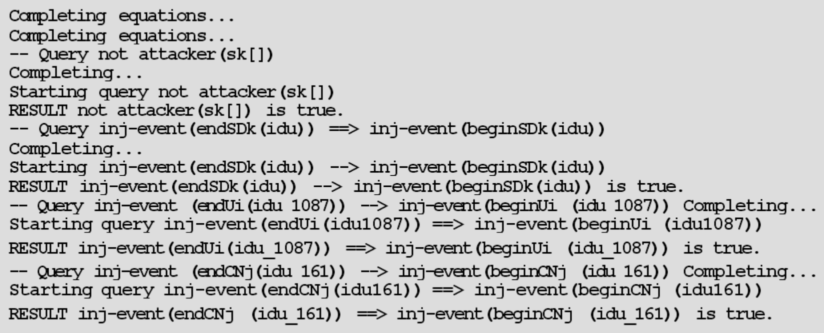
This sub-section presents the comparative analysis of the proposed scheme (SUSIC) with other contemporary schemes such as Chen et al. [22], Harishma et al. [32], Challa et al. [33], Ever et al. [34], Chen et al. [24] and Sutrala et al. [8]. We have performed simulation with the computation of execution timing for selected crypto-primitives using ”Multiprecision Integer and Rational Arithmetic Cryptographic Library (MIRACL)” [23] on the following two test beds.

- We deployed the first platform for server-based environment on Ubuntu 18.04.4 LTS, 2.8 GHz Intel Core i7-8565U, 16GB RAM, 64-bit CPU architecture. The execution experiment for each crypto-primitive was performed for 100 iterations for both platform settings on resource constrained IoT devices as well as server based environment.
- We deployed the second platform for CPS-based smart device or user mobile device having ”Raspberry PI 3 B+ Rev 1.3 having 64-bit processor, 1.4 GHz Quadcore with 4 cores, 1GB RAM, Operating system Ubuntu 2 0.04 LTS (64-bit edition)”.

To compute the execution timing for crypto-primitives, we assume that T_h, T_{ecm}, T_{eca}, T_{sym}, T_{fe}, T_{bp}, T_e, T_{IDe}, T_{IDd} characterize the timing for one-way hash function, elliptic curve point multiplication, elliptic curve point addition, symmetric encryption/decryption, fuzzy extractor, billinear pairing, modular exponentiation, identity-based encryption, identity-based decryption, respectively. The average run time of simulation for both platforms is used to compute the timing of crypto-primitives in milliseconds from 100 iterations. The timing of crypto-primitives is depicted in Table III.

1) Comparison of security functionalities: The Table II compares the security functionality comparisons ($SF_{C1} - SF_{C14}$) between SUSIC and other schemes [8], [22], [24], [32]–[34]. According to this table, the schemes [8], [24], [33] do not support mutual authentication. The schemes [22], [24], [32], [33] do not suspport either anonymity or untraceability. The schemes [24], [33] are prone to impersonation attack. Similarly the schemes [8], [24], [33] do not resist stolen verifier attack, replay attack, and de-synchronization attack, respectively. The schemes [24], [32] are prone to MITM attack, while [22], [32] suffer from ESL threat. Likewise, the protocols [24], [33] are prone to privileged insider attack. It is obvious that the proposed SUSIC is more effective in terms of security features $SF_{C1} - SF_{C14}$ in comparison with other schemes.

2) Computational costs: The proposed scheme SUSIC bears the computational costs of $9T_h + T_{eca} + 2T_{ecm} + 2T_{sym} \approx 12.1$ and $T_{fe} + 25T_h + 3T_{eca} + 9T_{ecm} \approx 22.647$ for controller node and smart device/mobile device, respectively as shown in Table IV.



```
Completing equations...
Completing equations...
-- Query not attacker(sk[])
Completing...
Starting query not attacker(sk[])
RESULT not attacker(sk[]) is true.
-- Query inj-event(endSDk(idu)) ==> inj-event(beginSDk(idu))
Completing...
Starting inj-event(endSDk(idu)) --> inj-event(beginSDk(idu))
RESULT inj-event(endSDk(idu)) --> inj-event(beginSDk(idu)) is true.
-- Query inj-event (endUi(idu 1087)) --> inj-event(beginUi (idu 1087)) Completing...
Starting query inj-event(endUi(idu1087)) ==> inj-event(beginUi (idu1087))
RESULT inj-event(endUi(idu_1087)) ==> inj-event(beginUi (idu_1087)) is true.
-- Query inj-event (endCNj(idu 161)) --> inj-event(beginCNj (idu 161)) Completing...
Starting query inj-event(endCNj(idu161)) ==> inj-event(beginCNj (idu161))
RESULT inj-event(endCNj (idu_161)) ==> inj-event(beginCNj (idu_161)) is true.
```

Figure 3: Verification using ProVerif code

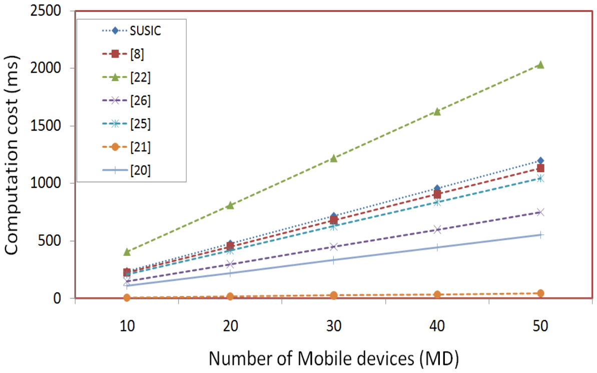


Figure 4: Number of MN nodes and computational cost

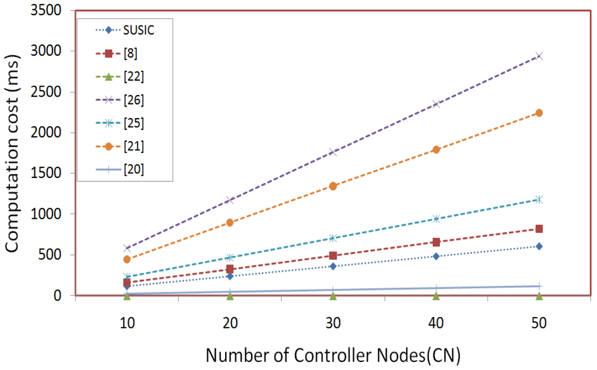


Figure 5: Number of CN nodes and computational cost

On controller/server node's end, the proposed scheme bears less computational cost of 12.1ms as compared to 16.49ms, 58.831ms, 23.675ms, 44.85ms incurred by [8], [34], [33], [32], respectively. On the end of smart/mobile device, the SUSIC bears less computational cost than [24] and almost equivalent to [8], however more than rest of the schemes [22], [32]–[34]. On the controller node's end, the computational costs are comparable with other schemes except [22]. The Fig. 4 and Fig. 5 show the impact of increasing number of mobile and controller nodes in the network. Although, the proposed scheme bears a little more computational cost than rest of the schemes as far as the smart or mobile device-based operations are concerned, the proposed scheme SUSIC is resistant to most of the known

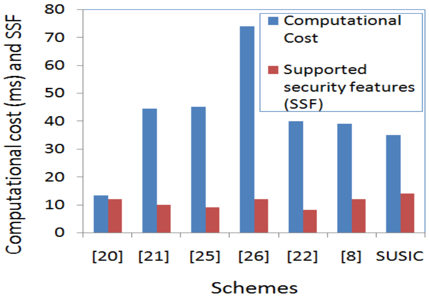


Figure 6: Supported security features and Computation

security threats unlike other schemes as shown in Table III and Fig. 6, which makes it even more suitable for practical implications in the wake of emerging security threats.

3) Communication cost: For comparing the communication costs of various schemes, we assume that the identity, hash digest, timestamp, P_x and P_y coordinates in elliptic curve point (P_x, P_y) take 160 bits, 160 bits, 32 bits and 320 bits with 160 bit each, respectively. In the proposed SUSIC model, the user U_i, controller node CN_j and smart device SD_k require to transmit the messages $\{M_{sgX} \mid 1 \leq X \leq 3\}$ with an aggregate 3040 bits of communication cost. The SUSIC communication cost is comparable with [8], [33]. Although, it bears a little more communication cost in comparison with rest of the schemes, yet the substantial advantage of supported security features tips the scale in favor of the contributed model as for practical implementation.

VII. Conclusion

The insecure communication-based hindrances has confined the evolution of cyber physical systems thus far. This article evaluates the security aspect of IIoT/CPS environment under the auspice of SDN-based controller node. To this end, a novel three-factor authenticated key exchange mechanism "SUSIC" has been designed for SDN-enabled IIoT/CPS network. The SUSIC ensures secure session key establishment between user and IIoT-oriented smart devices, through SDN controller entity. Our scheme supports mutual authenticity, anonymity and resistance against known attacks. Moreover, our scheme is supported with formal security analysis employing Real-or-Random (RoR) random oracle model. The performance evaluation demonstrates computational efficiency as compared to related schemes. Thus, SUSIC is more applicable and referential for future SDN-oriented IIoT/CPS projects.

References

[1] J. Franco, A. Aris, B. Canberk, and A. S. Uluagac, "A survey of honeypots and honeynets for internet of things, industrial internet of things, and cyber-physical systems," *IEEE Communications Surveys & Tutorials*, 2021.

[2] W. Lucia and A. Youssef, "A key-agreement scheme for cyber-physical systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021.

[3] K. Cao, S. Hu, Y. Shi, A. Colombo, S. Karnouskos, and X. Li, "A survey on edge and edge-cloud computing assisted cyber-physical systems," *IEEE Transactions on Industrial Informatics*, 2021.

[4] P. Krishnan, K. Jain, K. Achuthan, and R. Buyya, "Software-defined security-by-contract for blockchain-enabled mud-aware industrial iot edge networks," *IEEE Transactions on Industrial Informatics*, 2021.

[5] M. U. Hassan, M. H. Rehmani, and J. Chen, "Differential privacy techniques for cyber physical systems: a survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 746–789, 2019.

[6] A. M. Abdelrahman, J. J. Rodrigues, M. M. Mahmoud, K. Saleem, A. K. Das, V. Korotaev, and S. A. Kozlov, "Software-defined networking security for private data center networks and clouds: vulnerabilities, attacks, countermeasures, and solutions," *International Journal of Communication Systems*, vol. 34, no. 4, p. e4706, 2021.

[7] D. Chattaraj, S. Saha, B. Bera, and A. K. Das, "On the design of blockchain-based access control scheme for software defined networks," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 237–242, IEEE, 2020.

[8] A. K. Sutrala, M. S. Obaidat, S. Saha, A. K. Das, M. Alazab, and Y. Park, "Authenticated key agreement scheme with user anonymity and untraceability for 5g-enabled softwarized industrial cyber-physical systems," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[9] D. E. Sarmiento, A. Lebre, L. Nussbaum, and A. Chari, "Decentralized sdn control plane for a distributed cloud-edge infrastructure: A survey," *IEEE Communications Surveys & Tutorials*, 2021.

[10] N. Anerousis, P. Chemouil, A. A. Lazar, N. Mihai, and S. B. Weinstein, "The origin and evolution of open programmable networks and sdn," *IEEE Communications Surveys & Tutorials*, 2021.

[11] Y. Liu, Z. Laiping, J. Hua, W. Qu, S. Zhang, and S. Zhong, "Distributed traffic engineering for multi-domain sdn without trust," *IEEE Transactions on Cloud Computing*, 2021.

[12] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on information theory*, vol. 29, no. 2, pp. 198–208, 1983.

[13] R. Canetti and H. Krawczyk, "Universally composable notions of key exchange and secure channels," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 337–351, Springer, 2002.

[14] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE transactions on computers*, vol. 51, no. 5, pp. 541–552, 2002.

[15] S. A. Chaudhry, A. Irshad, J. Nebhen, A. K. Bashir, N. Moustafa, Y. D. Al-Otaibi, and Y. B. Zikria, "An anonymous device to device access control based on secure certificate for internet of medical things systems," *Sustainable Cities and Society*, vol. 75, p. 103322, 2021.

[16] M. Bilal and S.-G. Kang, "An authentication protocol for future sensor networks," *Sensors*, vol. 17, no. 5, 2017.

[17] M. C. SDK, "Multiprecision integer and rational arithmetic cryptographic library," 2020.

[18] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for industrial control systems: A survey," *computers & security*, vol. 89, p. 101677, 2020.

[19] E. Molina and E. Jacob, "Software-defined networking in cyber-physical systems: A survey," *Computers & electrical engineering*, vol. 66, pp. 407–419, 2018.

[20] J. M. Taylor and H. R. Sharif, "Security challenges and methods for protecting critical infrastructure cyber-physical systems," in *2017 International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT)*, pp. 1–6, IEEE, 2017.

[21] M. S. Chong, H. Sandberg, and A. M. Teixeira, "A tutorial introduction to security and privacy for cyber-physical systems," in *2019 18th European Control Conference (ECC)*, pp. 968–978, IEEE, 2019.

[22] Y. Chen, J.-F. Martinez, P. Castillejo, and L. Lopez, "A privacy protection user authentication and key agreement scheme tailored for the internet of things environment: Priauth," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.

[23] M. Bilal and S. Pack, "Secure distribution of protected content in information-centric networking," *IEEE Systems Journal*, vol. 14, no. 2, pp. 1921–1932, 2020.

[24] Y. Chen, J.-F. Martínez, P. Castillejo, and L. López, "A bilinear map pairing based authentication scheme for smart grid communications: Pauth," *IEEE Access*, vol. 7, pp. 22633–22643, 2019.

[25] M. Bilal and S.-G. Kang, "A secure key agreement protocol for dynamic group," *Cluster Computing*, vol. 20, no. 3, pp. 2779–2792, 2017.

[26] K. Renuka, S. Kumari, D. Zhao, and L. Li, "Design of a secure password-based authentication scheme for m2m networks in iot enabled cyber-physical systems," *IEEE Access*, vol. 7, pp. 51014–51027, 2019.

[27] P. T. Duy, H. Do Hoang, A. G.-T. Nguyen, V.-H. Pham, *et al.*, "B-dac: A decentralized access control framework on northbound interface for securing sdn using blockchain," *Journal of Information Security and Applications*, vol. 64, p. 103080, 2022.

[28] B. Alzahrani and S. A. Chaudhry, "An identity-based encryption method for sdn-enabled source routing systems," *Security and Communication Networks*, vol. 2022, 2022.

[29] L. Vishwakarma, A. Nahar, and D. Das, "Lbsv: Lightweight blockchain security protocol for secure storage and communication in sdn-enabled iov," *IEEE Transactions on Vehicular Technology*, 2022.

[30] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, S. Kumari, and M. Jo, "Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2884–2895, 2017.

[31] S. A. Chaudhry, A. Irshad, M. A. Khan, S. A. Khan, S. Nosheen, A. A. AlZubi, and Y. B. Zikria, "A lightweight authentication scheme for 6g-iot enabled maritime transport system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2401–2410, 2023.

[32] B. Harishma, S. Patranabis, U. Chatterjee, and D. Mukhopadhyay, "Poster: authenticated key-exchange protocol for heterogeneous cps," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pp. 849–851, 2018.

[33] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. G. Reddy, E.-J. Yoon, and K.-Y. Yoo, "Secure signature-based authenticated key establishment scheme for future iot applications," *Ieee Access*, vol. 5, pp. 3028–3043, 2017.

[34] Y. K. Ever, "A secure authentication scheme framework for mobile-sinks used in the internet of drones applications," *Computer Communications*, vol. 155, pp. 143–149, 2020.

[35] A. Irshad, S. A. Chaudhry, Q. Xie, X. Li, M. S. Farash, S. Kumari, and F. Wu, "An enhanced and provably secure chaotic map-based authenticated key agreement in multi-server architecture," *Arabian Journal for Science and Engineering*, vol. 43, no. 2, pp. 811–828, 2018.

[36] B. Blanchet, "Proverif automatic cryptographic protocol verifier user manual," *CNRS, Departement dInformatique, Ecole Normale Superieure, Paris*, 2005.