Received: October 7, 2024. Revised: October 29, 2024. 160

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

ChaoticPUFChain-IoT: PUF-based Authentication with Chaotic Maps for Blockchain-Enhanced IoT

Bharati B. Pannyagol[1]* **Santosh. L. Deshpande[1]**

[1]*Department of Computer Science and Engineering, Visvesvaraya Technological University, India*
* Corresponding author's Email: bharatibp13@gmail.com

Abstract: The expansion of communication technology has led to the hasty integration of the Internet of Things into our everyday lives. Because of their limited resources and uneven distribution, these devices are more open to attackers and other security risks. Therefore, to ensure their data security, a strong and trustworthy lightweight authentication mechanism is required. This paper proposes a lightweight authentication strategy based on Physical Unclonable Functions (PUF) and Chaotic map that provides two-way authentication between gateways and end devices. This solution uses a PUF and chaotic map for stronger security and lower resource overhead than traditional authentication techniques. PUF is utilized to generate Challenge Response Pairs without keeping any confidential authentication data in the end device's memory. Challenge –Response Pairs are protected while transmitting over insecure networks by using a Chaotic map. As a result, the proposed authentication process ensures security by being resistant to attacks including physical attacks, machine learning modeling, and impersonation attacks. The protocol's effectiveness is analysed and assessed using AVISPA and Scyther formal verification tools and the findings demonstrate that the proposed protocol successfully withstands 9 security properties such as two-way authentication, physical and cloning and user anonymity etc. Based on a comparative analysis the proposed scheme ensures security with minimal computing and communication costs, making it appropriate for resource-constrained environments.

Keywords: Authentication, Block-chain, Chaotic map, Internet of things, Physical unclonable functions, Security.

1. Introduction

The Internet of Things (IoT) is a network of physical objects, services, and information that can communicate, collaborate, and perform tasks outside of human interaction [1]. Global IoT devices will increase from 1.6 billion to 3.2 billion between 2020 and 2030, according to Gartner [2]. IoT devices are used in homes, transportation, and infrastructure etc. Security is crucial for IoT devices since they process sensitive data in inaccessible areas. Because of their limited processing power and memory, IoT devices struggle with energy efficiency [3-4]. Physical node assaults provide a security risk. For instance, if someone stole a node and duplicated its local data, they may appear as a genuine node or build many copies and expose them to the network. Early IoT security focused on passwords, encryption, and access restriction. As IoT systems get more complex, researchers understand they require a broader strategy. This method should include all IoT ecosystem components, not only devices and data processing/storage. Password-based methods are insecure for device-to-device mutual authentication. For safe IoT systems, cryptography is recommended. But IoT nodes lack encryption capabilities due to storage and processing needs. It drains IoT nodes' batteries and reduces their lifespan.

Given these memory and processing restrictions on devices as well as energy conservation, most studies have focused on lightweight symmetric ciphers. Contrast this with cryptography, using keys that must be both secret from others and each other while also being concealed thus obscure and difficult to break. Physical tampering and illegal access must be avoided by keeping keys within a secure boundary. This is where the hardware used to protect keys from physical attacks and data loss can often be found.

Physically Unclonable Functions (PUF) provide safe key creation without storage. PUFs profit from the intrinsic unpredictability of the IC manufacturing and fabrication process. This approach makes PUFs unique; i.e two PUFs cannot be identical [5]. This has motivated researchers to explore affordable PUF solutions for constrained devices.

The IoT community has warmed up to the idea of employing chaotic maps as a new method for identifying devices [6]. Since chaotic maps are scientific functions with complex and unpredictable behavior, they are ideal for use in cryptography. It is possible to employ chaotic maps within the framework of the IoT, to design robust reliable authentication methods that account for the unique constraints of IoT devices [7, 8].

These techniques support, but centralized IoT authentication solutions still use an underlying infrastructure and have a single point of failure. If centralized authority is challenged, safety concerns may spread quickly. Conversely, IoT architecture consists of a federation or ecosystem of several platforms, networks, and systems, each owned by a different corporation. It is not easy or practical to impose a single source of trust or centralized authority within an IoT ecosystem. To address this risk associated with centralization, Blockchain technology has attracted a lot of attention.

IoT devices may create safe digital identities and identify themselves without a central authority using BC technology's decentralization and tamper-resistance [9, 10]. This approach addresses the challenges of resource-constrained IoT devices, as the authentication process is distributed across the network, reducing the burden on individual devices [11]. The distributed consensus mechanism and immutable record-keeping capabilities of BC, IoT devices can establish trust and securely authenticate with cloud servers, base stations, and other network entities [11]. Additionally, the use of BC can enhance the transparency of the system, allowing for the immediate detection of any unauthorized modifications of data captured by IoT devices [12].

Blockchain immutability ensures data integrity and builds trust between communication partners. True randomisation of chaotic maps and secure key generation without storage of CRPs in PUF establish the safe architecture for blockchain-based IoT devices.A novel authentication mechanism and secure communication implemented using these traits.

Rest of paper outline: In Section 2 and 3, we discuss PUF-based authentication in IoT and its history and current research. The suggested solution is discussed detailed in Section 4. Section 5 gives the protocol's crypt analysis, whereas Section 6 gives its performance analysis. The work concludes in section 7.

2. Scientific background

2.1 Physical unclonable functions

As mentioned earlier, PUFs are used in this work to reinforce device perimeters and apply security measures. To be precise, PUFs are used as safe key producers.

Before we delve into our approach, we offer a brief gestalt of the essential properties of PUFs. In 2001, Pappu [13] presented PUFs as a hardware security primitive for silicon authentication. The development of PUFs has emerged as a promising approach to enhance hardware security and trust in IoT [14]. PUFs are innovative security primitives that leverage the inherent manufacturing variations within electronic devices to generate a unique secure digital fingerprint, providing a robust, cost-effective, secure, and reliable means of device authentication and key generation [15]. The hardware-based approach to PUF involves the incorporation of specialized circuit structures, such as SRAM PUFs, Arbiter PUFs, and Ring oscillator PUFs, into electronic devices[16]. These PUF circuits exploit the inherent randomness in the manufacturing process to produce unique, unpredictable responses to input challenges, effectively creating a secure fingerprint for the device. Efforts to realize software-based PUFs have also been explored, with techniques like virtual secure co-processing and information-flow control enabling the creation of secure software vaults on platforms like Android [17, 18].

PUFs are functions that take challenges as input and create random yet device-specific replies [19]. When a device is challenged with a specific input, the PUF generates a unique response that can be verified by the authenticating authority. Every device is guaranteed to have a unique digital identity via this Challenge-Response (CRP) method, thwarting illegal access and certifying the integrity of the IoT ecosystem. Strong PUFs and Weak PUFs are the two types of PUFs [20, 21].

2.2 Chaotic map

The enactment of chaotic maps in authentication mechanisms for IoT devices has gained substantial consideration in recent years. Chaotic systems are a viable way to improve the security of IoT settings because of their unexpected behaviour and sensitivity to beginning circumstances [22]. By leveraging the inherent complexity and unpredictability, researchers

have developed authentication protocols that can effectively mitigate the risks associated with traditional password-based systems, which are vulnerable to various attacks [23]. The integration of chaotic map-based techniques in IoT device authentication processes has demonstrated improved resilience against threats such as replay attacks, MiTM attacks, and brute-force attempts, contributing to the overall security and trustworthiness of IoT ecosystems [4].

A unique chaotic sequence was suggested by the author [24] which is written in Eq. (1).

$$y_{j+1} = cos\left(\pi\left(G\left(b, y_j\right) + H\left(c, y_j\right) + \gamma\right)\right) \quad (1)$$

Here, $G\left(b, y_j\right)$ and $H\left(c, y_j\right)$ are seed sequence sets, and γ represent a changing constant, b and c are control parameters. Eq. (1) shows that our model knows the outcome of $H\left(c, y_j\right)$and $G\left(b, y_j\right)$with γ . Between the two seed sequence sets, the function helps to effectively shuffle their chaotic sequence dynamics. Additionally, attaining high complex nonlinearity and the service is facilitated by the use of cosine transformation.

2.3 Blockchain technology

BC technology records transactions in a distributed, decentralized digital ledger using networked computers [25-26]. Its key qualities are transparency, immutability, and security. Each BC's block contains a unique cryptographic hash that ties it to the block preceding it, establishing an irrevocable sequence, so everyone can observe and verify BC transactions [27] . This ensures integrity of data, as any attempt to modify a previous transaction would be detected by the network. Furthermore, the decentralized nature of BC eliminates the requisite for a principal authority, making the system resistant to single points of failure or control [28]. This method might revolutionize IoT device identification verification. BC decentralization and immutability allow businesses to create a transparent and secure framework for IoT device validation. In paper[29] author defines technique that eliminates administration authentication concerns, boosting IoT network security and reliability. BC and IoT integration is predicted to spread across numerous industries. This ensures connected devices work reliably and securely.

3. Literature survey

Node-to-node topologies have seen various research on low-resource device authentication techniques. The author in [30] used Elliptic Curve Cryptography and PUF to implement identity based authentication but it does not ensure the PUF's Challenge Response Secrecy and have high computational time complexity. In paper [31] author uses Elliptic Curve Diffie Hellman protocol for authentication but it focuses only mutual authentication between devices, rather than device-to-server authentication and computation cost of ECDH algorithm is high. Elliptic curve encryption, fuzzy extractor, and PUF were used to provide a lightweight authentication method for IoT devices and neighbourhood gateways to safeguard sensitive data in [32]. The approach uses dot multiplication Elliptic curve cryptography and fuzzy extractors, are too expensive for low-resource systems. In [33], device secret values or user-generated values determine the authentication value. Physical attacks on devices may compromise sensitive information, which can be exploited for impersonation or man-in-the-middle attacks. Author in [34] proposed a lightweight device-to-device mutual authentication technique for vehicle-to-roadside unit systems using hash, PUF, fuzzy extractor, and Chebyshev chaotic map. Unencrypted data transmitted between devices is vulnerable to machine learning modelling attacks. The [35] proposed smart grid authentication solution employs PUF, hash, and Chebyshev chaotic map but is computationally and communication ally costly & inappropriate for low-resource devices. Fuzzy extractors and PUF were used for authentication in [36]. Logging onto devices requires entering a password, which isn't applicable to equipment situated in distant areas. Fuzzy extractor require high computation cost.

3.1 Contribution

This work proposes a PUF and chaotic map-based, lightweight authentication technique for IoT devices with low power resources to address the above challenges. The main advantages of this study are:

1. Our IoT device authentication is straightforward and anonymous. This method allows mutual authentication between the gateway and terminal device without user input of biometric or password information. Instead of saving CRPs, devices produce unique IDs utilizing PUF technology for mutual authentication. Using the chaotic map to secure data transported over insecure networks protects private data like CRPs from physical and machine learning threats.
2. A system with minimal computational and communication costs that provides anonymity, un-traceability, forward/backward

confidentiality, etc. is proposed in this study. A comparison examination of security performance, computational cost, and communication cost with authentication techniques in [32-36] suggests it is appropriate for resource-constrained devices.

4. Proposed solution

Specifically for the IoT, PUF can enhance the security guarantees of an authentication protocol. Because IoT nodes are vulnerable to hacking and physical capture, there is a chance that stored secrets could leak. When it comes to reducing this vulnerability, using a PUF is essential. So here we are proposing the framework based on Challenge Response Protocol (CRP) to authenticate IoT device and Gateway Node(GW). Initialization, registration, and authentication 3 basic steps comprise this protocol. Figs. 2 and 3 depict the whole procedure.

4.1 System model

The suggested framework is designed and analysed using these network and threat models:

4.1.1. Network model

The suggested framework's network model is shown in Fig. 1. The BC connects gateway servers and IoT devices. The device layer is made up of several IoT gadgets that are utilized in smart homes, everything around them despite having limited processing and storage capacity. Through wireless wind farms, smart factories, medicals, and smart thermostats. These gadgets must gather data from connectivity, the closest GW gathers this data i.e subsequently sent to cloud servers through BC for long-standing storage. This design links the GW to the BC to register and authenticate IoT devices. These servers mine and add blocks in the proposed BC. All of the communication, takes place over the Internet, an insecure open route i.e vulnerable to several threats. Thus, we propose a ChaoticPUFChain-IoT method.

4.1.2. Threat model:

The suggested framework was based on the popular "Dolev-Yao (DY model)"[37]. GW and IoT devices communicate over an unprotected open channel. Thus, an attacker, "A," has a greater probability of exploiting shared data for harmful purposes. Due to security vulnerabilities, attacker "A." might leak, delay, change, or erase data transferred between IoT devices and GW. Using the information, "MiTM attacks," "impersonation attacks," "credentials guessing attacks," and "unlawful session key computation attacks" may be performed.

4.2 Authentication mechanism

Two main phases of the proposed framework, each of which aims to provide safe communication between the different parties. These stages consist of setting up the framework parameters in the beginning, registering the entities, and authenticating the interactions between the entities. Table 1 will list the symbols and their respective parameter description used here.

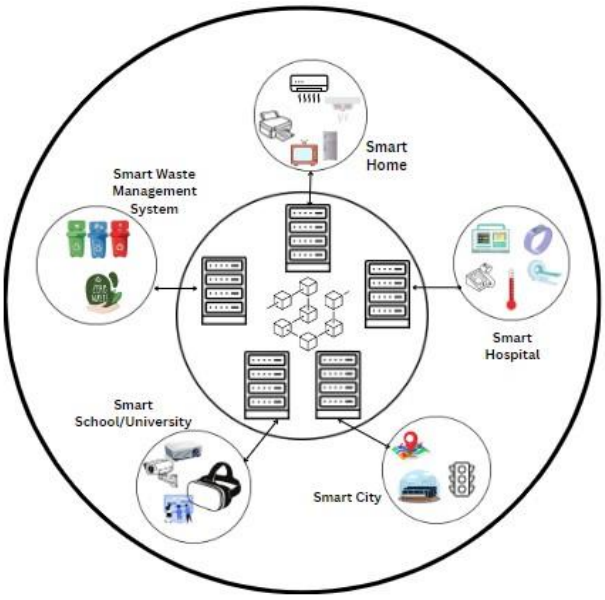

Figure. 1 Network Model

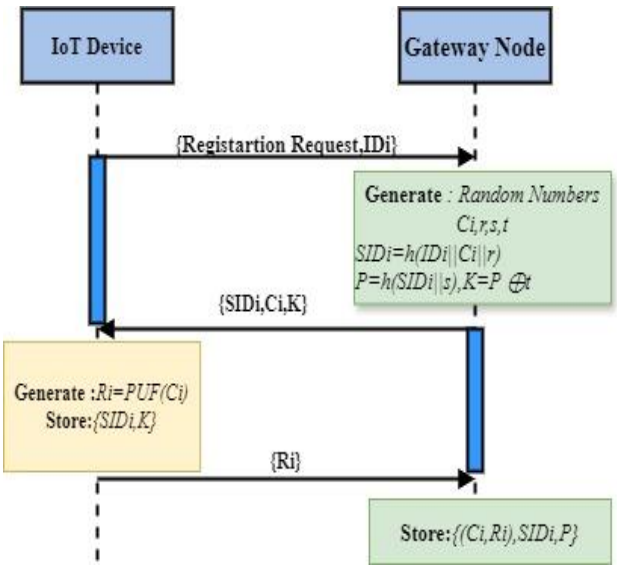

Figure. 2 Registration Phase

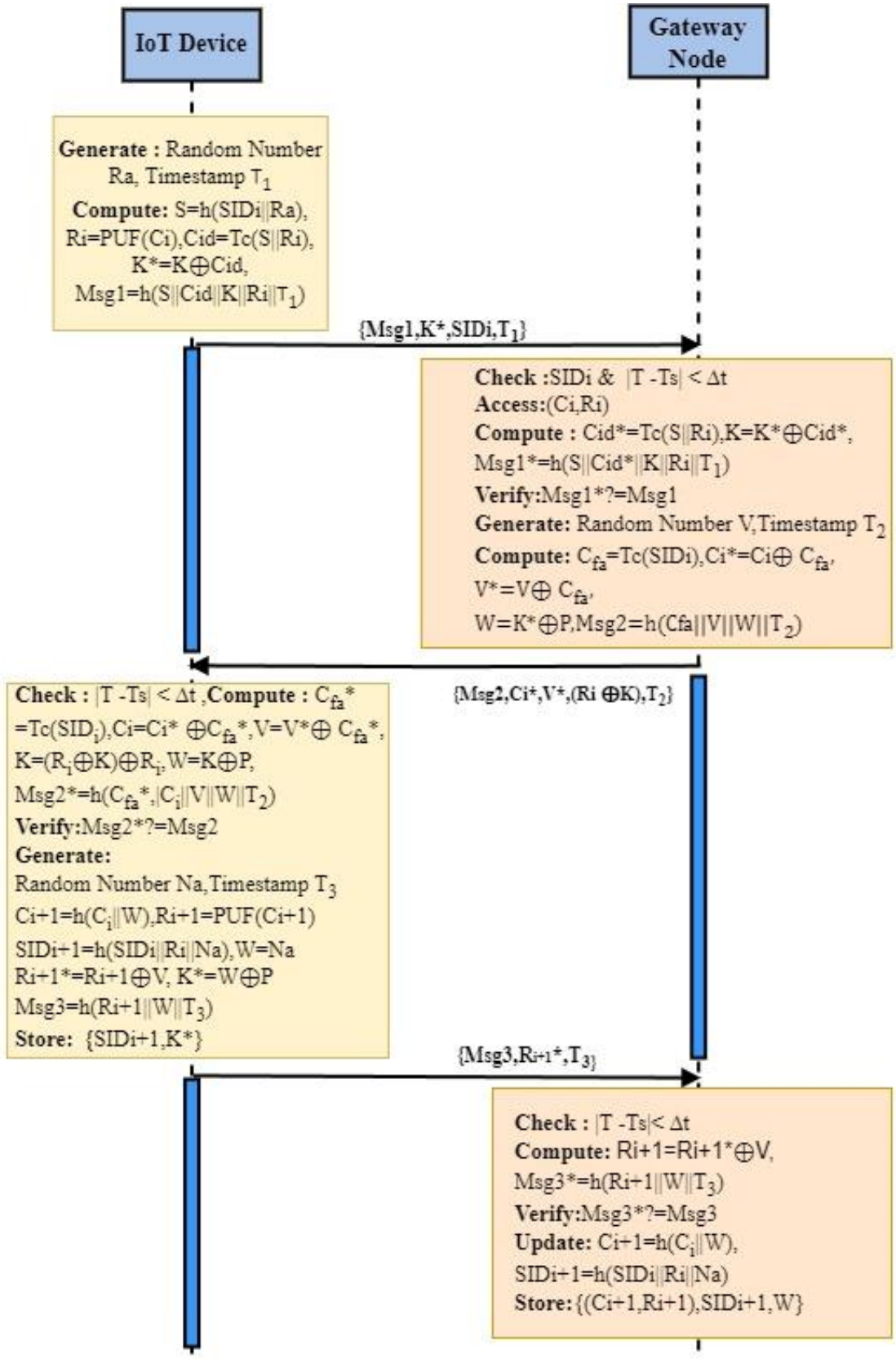

Figure. 3 Authentication Phase

Table 1. Symbols and Parameters Description

Symbols	Description
Di, FAi	Device, Fog Node
SIDi	Pseudo Identity of IoT Devices
IDi	IoT device Identification Number
(Ci,Ri)	PUF Challenge Response Pair
h(.)	Hash Function
r,s,t,u,V,p	Pseudo-Random Numbers
Ti, i=1,2,3…	Timestamps
Δ, Δij Ti-Tj	Time Interval
Tc (x)	Novel-chaotic-function
PUF(.)	Physical Unclonable Function
⊕ and ||	XOR and Concatenation function

4.2.1. Registration phase

The IoT device and the GW registration are presented in Fig.2. To get the required authentication parameters, the device Di and GW communicate via a secure private channel. The following are the precise steps involved in registering:

Step 1:
IoT device Di sends its real identity and request to GW. {IDi, Request}

Step 2:
GW generates r,s and t € Zq* randomly secret parameters, which will be updated periodically, then selects the challenge Ci and computes the Pseudo Identity **SIDi=h(IDi||Ci||s).**
Then compute the shared secrets **P=h(SIDi||s)** and **K=P⊕t**. GW sends the message **{SIDi,Ci,K}** to Di

Step 3:
Di get the message, stores **{SIDi,K}** into the device, then calculates the PUF response Ri for challenge Ci and sends the message **{Ri}** securely to the GW

Step 4:
GW stores **{SIDi,(Ci,Ri),P}** into the BC.

4.2.2. Authentication phase:

The authentication phase is presented in Fig. 3. Two-way authentication is conducted by the gateway and the terminal device using the authentication parameters that were acquired during registration.

Step 1:

Di generates the Random number Ra, Timestamp T1 then computes

S=h(SIDi|Ra), Ri=PUF(Ci), Cid=Tc (S||Ri),

K*=K ⊕ Cid, Msg1=h(S||Cid||K||Ri) then generate the authentication message {Msg1,S,K*,SIDi,T1} and send to GW.

Step 2:

When GW receives an authentication request {Msg1,S,K*,SIDi,T1} from Di, GW determines whether the transmission delay, or |T -Ts| < Δt, is less than Δt. If it is, the authentication process is proceed

Then it will check the database for SIDi; if SIDi is not there, GW will reject it. Di will then start the authentication request process again.

Step 2.1:

Meanwhile, GW will access records from the Block-chain {SIDi,(Ci,Ri),P}. And computes

Cid'= Tc (S||Ri), K=K*⊕Cid'

Msg1'= h(S||Cid'||K||Ri)

if Msg1= =Msg1' then GW authenticate the Device Di.

Step2.2

GW generates the Random number V, Timestamp T2 And Calculates

Cfa= Tc(SIDi), Ci*=Ci⊕Cfa, V*=V⊕Cfa

W=K*⊕P, Msg2 =h(Cfa||V||W)

GW Generate message {Msg2,Ci*,V*,(Ri⊕K),T2} & send it to device Di

Step 3: Authentication at device side

Step 3.1

After receipt the message {Msg2,Ci*,V*,(Ri ⊕ K),T2},Di determines whether the transmission delay, or |T -Ts| < Δt, is less than Δt. If it is, the authentication process proceeds.

Step 3.2

Di computes Msg2'

Cfa'= Tc (Cid) ,Ci=Ci*⊕Cfa', V=V*⊕Cfa', K=(Ri⊕K)⊕Ri,W=K⊕P, Msg2'=h(Cfa'||Ci||V||W)

Compare the calculated Msg2' with received Msg2 i.e Msg2= =Msg2' if both are equal then it authenticates the GW.

Step 3.2

Generate the Random Number Na, Timestamp T3 Ci+1=h(Ci||W), Ri+1=PUF(Ci+1), K*=W⊕P, SIDi+1=h(SIDi||Ri||Na), Ri+1*=Ri+1⊕V, Msg3=h(Ri+1||W)

Send the message {Msg3, Ri+1*,T3} to GW

Store :{ SIDi+1, K*}

Step 4: When FA receives an authentication request {Msg3, Ri+1*, T3} from Di, GW determines whether the transmission delay, or

|T -Ts| < Δt, is less than Δt. If it is, the authentication process is proceed

Compute: Ri+1= Ri+1*⊕V, Msg3'=(Ri+1||W)

Compare the calculated Msg3' with received Msg3 i.e Msg3= =Msg3' if both are equal then Update the Data and store in Block-chain.

Update: Ci+1=h(Ci||W), SIDi+1=h(SIDi||Ri||Na)

Store :{(Ci+1, Ri+1) ,SIDi+1,W

5. Crypt analysis

5.1 Informal verification

5.1.1. Two way authentication

This article offers a two-way device-gateway authentication method. The gateway authenticates the device by verifying Msg1* = = Msg1 and the device authenticates the gateway by checking Msg2*= =Msg2. The chaotic map discrete logarithm problem will arise while trying to obtain Cid and Cfa from Msg1, Msg2, and Msg3, which include secret values like S, Cid, Ri, Cfa, V, and W. Attackers cannot utilize V, Ri+1, and W as terminal devices or participate in gateway authentication since their encrypted values cannot be read directly.

5.1.2. Anonymity and un traceability

During the authentication process, both the device and the gateway employ pseudo-identity, which is then updated after each authentication. The GW and IoT devices' private parameters, p,r,s,t,u and V, are kept secret during the authentication procedure. This protocol efficiently ensures the anonymity of the system by making it difficult for attackers to trace pseudonyms and secret parameters due to their dynamic nature.

5.1.3. Resistance to replay attacks

By including timestamps to verify if the transmission delay satisfies the criteria prior to authentication, the proposed approach prevents the attacker from launching a replay attack through message resending. Furthermore, this approach further appends timestamps to Msg1 Msg2 and Msg3. Hence, authentication will fail if the attacker starts an attack by manipulating timestamps. As a result, the suggested approach is resistant to replay attacks since the secret values in Msg1 and Msg2 and Msg3 will be changed following each authentication.

5.1.4. Resistance to tamper attacks

This paper proposes a scheme that protects data exchanged during authentication with hash functions or XOR operations, preventing attackers from accessing secret values.

5.1.5. Physical and cloning attacks

In this scheme, every IoT device has an inbuilt PUF module. Any physical manipulation on the part of the device done by attacker, will change the PUF's physical properties, which will modify its output. Even with the same input, multiple PUF modules will produce distinct outputs because of the PUF's indestructibility and uniqueness. Furthermore, PUFs are physically unclonable, as covered in section 2.1, which makes this system resistant to both cloning and physical attacks.

5.1.6. Resistance to machine learning modeling attacks

The attacker develops a PUF response model using ML algorithms and the collected CRPs to predict future CRPs. In this scheme, the attacker is limited to capturing the CRPs from the insecure channel. Thus to obtain the challenge value Ci needs to obtain Cid first, the calculation of Cid will face the problem of the chaotic map, the attacker is unable to obtain Ci. The response value is hashed by the hash function, as the hash function is one-way, attacker cannot obtain Ri. As a result, the attacker is unable to gather the CRPs, making the suggested technique immune to attacks using ML modeling.

5.1.7. Resistance to DoS attacks

In the case that an adversary sends a torrent of useless data to disrupt connection, the device and gateway will confirm the transmission delay before examining Msg1, Msg2 and Msg3. If it don't meet any of the requirements, authentication will be prevented.

5.1.8. Resistant to impersonation attacks

Attackers must transmit the proper SIDi, K*, Msg1, Msg3, and Ri+1* to the gateway in order to pose as a genuine device. But in order to generate the right Msg1, the correct Cid,S,K, and Ri+1 are needed. It is obvious from the study above that the attacker is unable to get the right Cid,S,K, and Ri+1. As a result, the attacker is unable to use a false device identity to authenticate with the gateway. Attackers must have the proper CRPs in order to transmit Msg2,Ci*,V*, and (Ri $\oplus$ K) to the device in order to pose as a gateway. It's obvious from the study above that the attacker is unable to get the right CRPs, Cfa and Cid. As a result, the attacker cannot pose as a trustworthy gateway and use the device to authenticate.

5.1.9. Forward/backward security

W, Ri+1, and Cid in the suggested method will be changed following each authentication, thus even if an attacker manages to get their hands on the device's secret values and CRPs, they will be unable to follow the device's communication data from the past and future. Consequently, this paper's suggested system offers both forward and backward security.

```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 TYPED_MODEL

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

GOAL
 As Specified

BACKEND
 CL-AtSe

STATISTICS

 Analysed  : 31 states
 Reachable  : 21 states
 Translation: 0.01 seconds
 Computation: 0.00 seconds
```

Figure. 4 CL-ATSe Output

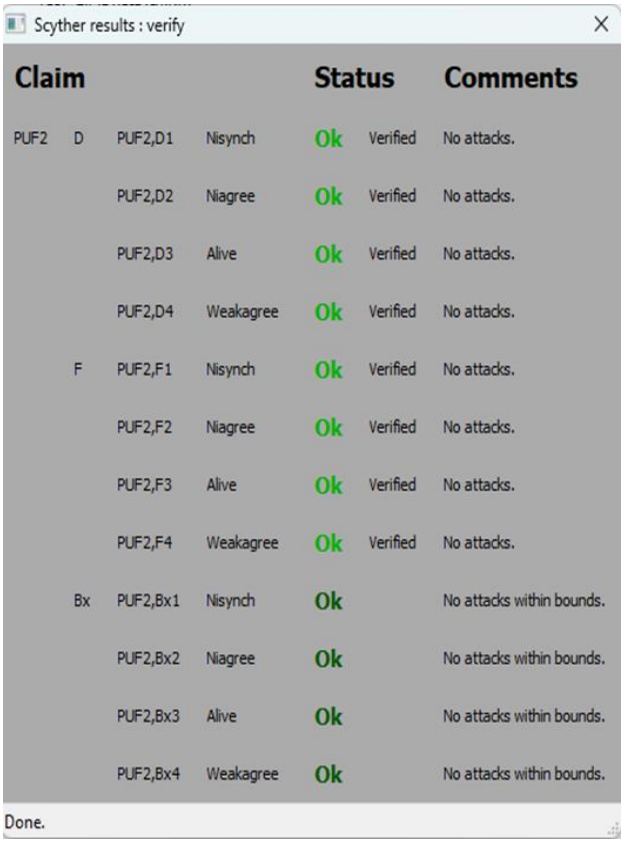

Figure. 5 Scyther Tool Output

5.2 Formal verification

5.2.1. Using AVISPA tool

The famous software AVISPA is used to test the suggested framework [38]. We defined our protocol in HLPSL utilizing the AVISPA tool for CL-AtSe back ends testing. HLPSL has three roles: D agent, FA agent, and Block agent, along with session, environment, and aim. The HLPSL implementation framework addresses registration and authentication.

The Fig. 4 demonstrate the finding of the CL-AtSe back end. In CL-AtSe back ends it analyzed 31 states out of which 21 states are reachable.

5.2.2. Using Scyther tool

Scyther tool [38] is used to formally verify security methods, with an emphasis on authentication in particular. It models possible attack situations, enabling protocol correctness analysis and verification. Properties like truthfulness, authenticity, and secrecy can all be confirmed by Scyther. Fig. 5 shows the output of the proposed scheme in Scyther tool.

6. Performance analysis

In this paper, we utilize the ZYNQ7000 series FPGA development board to simulate the terminal device. This board features a dual-core ARM Cortex-A9 processor running at 767 MHz and 1 GB of RAM. To simulate the gateway, we employ a system powered by a Core-i5 processor at 2.5 GHz with 16 GB of RAM. Various operations are implemented using the OpenSSL library. Compare this protocol's performance against other IoT authentication techniques. Comparing security attributes across protocols highlights our solution's benefits. We call the time needed for a 1 PUF answer T_{PUF}, the hash operation T_h, and the chaotic function T_{che}. Table 2 Shows the Cryptographic Operations and their execution times in microseconds (µs).

Table 3 shows the operation on the device side and the gateway side also a comparison of the computational expenses in µs of the various strategies and the scheme.

Literature [32, 33] use resource-intensive fuzzy extractor functions, which lead to the highest computational costs of 3549.83 and 3909.22, respectively. Literature [34] also uses the multiplier point operation on the ECC and fuzzy extractor,

Table 2. Cryptographic Operations Execution Cost

Cryptographic Operations	Exe. Time /µs
T_{PUF}	5.60
T_h	2.52
T_{che}	83.02

Table 3. Comparison of Execution Cost

Scheme	Device Side	Gateway Side	Total Time µs
[32]	5Th+TFE.Rep+5TMul+2TPUF	4Th +4TMul	3549.83
[33]	11Th+TFE.Rep+6TMul	6Th +6TMul	3909.22
[34]	3Th+TFE.Rep+2Tche+2Ts+TPUF	5Th +2Ts+2Tc	21536
[35]	7Th+3Tche+2TPUF	6Th +2Tche	328.41
[36]	11Th+4Tc+5Ts	3Th+2Tc +Ts	22816
Proposed Scheme	6Th+Tche+3TPUF	6Th +2Tche	269.16

leading to the higher computation cost of 21536 literature [35]. While circumventing dot multiplication operations on elliptic curves, leads to a cost of 328.41. Literature [36] led to the computation cost of 22816. Proposed Scheme will take less computational time compare to other schemes.

6.1 Communication cost

Prior to comparing communication costs, we assume fixed lengths for the following data elements: pseudo identity (128 bits), CRPs (128 bits), nonce (64 bits), symmetric encryption/decryption output (128 bits), hash function output (128 bits), elliptic curve dot multiplication output (256 bits), chaotic map output (128 bits), timestamps output (32 bits).
And modulo power operations output length (128 bits).

Table 4 summarize the total no. of messages exchanged between Device and GW and comparison with other literatures. From Table 4 we can conclude that our scheme communication cost is very less i.e 1056 bits.

Table 4. Communication Cost

Scheme	No. of Messages	Communication Cost in bits
[32]	3	1472 bits
[33]	2	2304 bits
[34]	4	1921 bits
[35]	3	1216 bits
[36]	4	1344 bits
Proposed Scheme	3	1056 bits

6.2 Security feature comparison

Machine learning modeling attacks may target CRPs obtained by eavesdropping, impersonation, and other means in literatures [30, 34, 36]. This work proposes a strategy that strengthens its resistance against machine learning modeling assaults by preventing attackers from gaining CRPs, as shown in the prior research. Literatures [30-32] use computationally demanding ECC and ECDH authentication. The chaotic map is used in this study. About a third of ECC computation expenses are chaotic maps. In literature [33] the authentication value is calculated using the device's secret values or the user's temporarily produced random values, therefore impersonation and man-in-the-middle attacks may destroy authentication if the device is attacked physically and secret information leaks. Previous study deemed the paper's approach secure against man-in-the-middle and impersonation attacks.

7. Conclusion

A lightweight authentication mechanism is proposed in this paper it leverages the security advantages of PUF and Chaotic map to provide a robust two-way authentication process between gateways and end devices. PUF eliminate the need to store sensitive authentication data in device memory and Chaotic map protect critical information transmission in public channels. The formal verification tools AVISPA and Scyther are used to analyze the results and the proposed scheme results shows that it satisfies 9 security features like two-way authentication, physical and cloning and user anonymity, with minimum computational cost of 269.16 µs and communication cost of 1056 bits. Comparative analysis with existing authentication schemes shows that the proposed papers authentication technique outperforms others in two-way authentication, user anonymity, and other security characteristics. Its minimal processing and communication overheads decrease resource usage and make it suitable for resource-constrained applications. In the future we can focus on testing the effectiveness and scalability of the suggested approach in real IoT environment.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, Investigation, Methodology, Validation, Formal Analysis, Writing-Original Draft

preparation, Bharati Pannyagol; Conceptualization Supervision, Methodology, Formal Analysis, investigation, Writing-review & editing, S. L. Deshpande.

References

[1] K. Wang, Y. Wang, Y. Sun, S. Guo, and J. Wu, "Green Industrial Internet of Things Architecture: An Energy-Efficient Perspective", *IEEE Commun. Mag.*, Vol. 54, No. 12, pp. 48-54, 2016.

[2] Neebal - Best Services, "IoT Consulting Service Provider | IOT Solutions | Neebal Technologies", Neebal - Best Services, 2024.

[3] P. Morgner, S. Pfennig, D. Salzner, and Z. Benenson, "Malicious IoT Implants: Tampering with Serial Communication over the Internet", *Research in Attacks, Intrusions, and Defenses*, in *Lecture Notes in Computer Science*, Vol. 11050. pp. 535-555.

[4] B. Pannayagol and S. Deshpande, "Security in Internet of Things: An Overview", In: *Proc. of International Conf. on DICCT*, Dehradun, India, pp. 243-248, 2023

[5] J. Panicker, A. Salehi, and C. Rudolph, "Authentication and Access Control in 5G Device-to-Device Communication", In: *Proc. of International Conf. on Trust Com*, Shenyang, China, pp. 1575-1582,2021

[6] L. Sun and Q. Du, "A Review of Physical Layer Security Techniques for Internet of Things: Challenges and Solutions", *Entropy*, Vol. 20, No. 10, p. 730, 2018,

[7] B. Pannayagol, S. Deshpande, and S. Yadav, "A Survey on a Security Model for the Internet of Things Environment", *Social Engineering in Cybersecurity*, 2024, pp. 171-186, 2024.

[8] F. Restuccia, "Blockchain for the internet of things: Present and future", *IEEE Internet of Things Journal*, Vol. 1, No. 1, pp. 1-8, 2018.

[9] B. Pannyagol and S. Deshpande, "Authentication in Blockchain-Based IoT Devices: A Review", In: *Proc. of 2024 International Conf. on ISCS*, Gurugram, India, pp. 1-5, 2024.

[10] Z. Tian, B. Yan, Q. Guo, J. Huang, and Q. Du, "Feasibility of Identity Authentication for IoT Based on Blockchain", *Procedia Computer Science*, Vol. 174, pp. 328-332, 2020,

[11] G. Rathee, F. Ahmad, N. Jaglan, and C. Konstantinou, "A Secure and Trusted Mechanism for Industrial IoT Network Using Blockchain", *IEEE Trans. Ind. Inf.*, Vol. 19, No. 2, pp. 1894-1902, 2023,

[12] H. Kim and E. A. Lee, "Authentication and Authorization for the Internet of Things", *IT Prof.*, Vol. 19, No. 5, pp. 27-33, 2017,

[13] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical One-Way Functions", *Science*, Vol. 297, No. 5589, pp. 2026-2030, 2002.

[14] P. H. Griffin, "Security for Ambient Assisted Living: Multi-factor Authentication in the Internet of Things", In: *Proc. of 2015 IEEE Globecom Workshops*, pp. 1-5, 2015.

[15] M. Tehranipoor and C. Wang, Eds., "Introduction to Hardware Security and Trust", *Springer Science & Business Media*, pp. 65-102, 2011.

[16] R. K. Shyamasundar, N. V. Narendra Kumar, and P. Teltumde, "Realizing software vault on Android through information-flow control", *IEEE Symposium on Computers and Communications*, pp. 1007-1014,2017.

[17] J. P. McGregor and R. B. Lee, "Protecting cryptographic keys and computations via virtual secure coprocessing", *SIGARCH Comput. Archit. News*, Vol. 33, No. 1, pp. 16-26, 2005.

[18] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, "Behavioral Fingerprinting of IoT Devices", In: *Proc. of the 2018 Workshop on Attacks and Solutions in Hardware Security*, Toronto Canada, pp. 41-50, 2018.

[19] M. O'Neill, "Insecurity by Design: Today's IoT Device Security Problem", *Engineering*, Vol. 2, No. 1, pp. 48-49, 2016,

[20] M. Schuß, J. Iber, J. Dobaj, C. Kreiner, C. A. Boano, and K. Römer, "IoT Device Security the Hard(ware) way", In: *Proc. of the 23rd European Conf. on Pattern Languages of Programs*, Irsee Germany, pp. 1-4, 2018.

[21] Y. Xie, G. Li, P. Wang, and Z. Zhou, "A compact weak PUF circuit based on MOSFET subthreshold leakage current", *IEICE Electron. Express*, Vol. 19, No. 21, pp. 20220415-20220415, 2022,

[22] W. E. H. Youssef *et al.*, "A Secure Chaos-Based Lightweight Cryptosystem for the Internet of Things", *IEEE Access*, Vol. 11, pp. 123279-123294, 2023,

[23] B. B. Pannyagol and S. L. Deshpande, "9 A Survey on Issues in Integrating Blockchain and IoT Technologies", *Blockchain for IoT Systems: Concept, Framework and Applications*, p. 123, 2024.

[24] S. B. K. and R. G. K., "An efficient data masking for securing medical data using DNA

encoding and chaotic system", *IJECE*, Vol. 10, No. 6, p. 6008, 2020,
[25] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", *Satoshi Nakamoto Institute*, 2008.
[26] N. Radziwill, "Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World", *Quality Management Journal*, Vol. 25, No. 1, pp. 64-65, 2018.
[27] D. Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. *Berkeley, CA: Apress*, 2017.
[28] N. Kshetri, "Can Blockchain Strengthen the Internet of Things?", *IT Prof.*, Vol. 19, No. 4, pp. 68-72, 2017.
[29] M. Díaz, C. Martín, and B. Rubio, "State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing", *Journal of Network and Computer Applications*, Vol. 67, pp. 99-117, 2016.
[30] A. Braeken, "PUF Based Authentication Protocol for IoT", *Symmetry*, Vol. 10, No. 8, p. 352, 2018.
[31] H. Akhundov, E. van der Sluis, S. Hamdioui, and M. Taouil, "Public-Key Based Authentication Architecture for IoT Devices Using PUF", In: *Proc. of 6th International Conf on Computer Science, Engineering and Information Technology (CSEIT-2019)*, pp. 353-371, 2017.
[32] H. D. Bai, "A smart grid device authentication scheme based on physically unclonable functions", *J. South-Cent. Univ. Natl., Nat. Sci. Ed.*, Vol. 42, No. 3, pp. 382-386, 2023.
[33] P. Soni, J. Pradhan, A. K. Pal, and S. H. Islam, "Cybersecurity Attack-Resilience Authentication Mechanism for Intelligent Healthcare System", *IEEE Trans. Ind. Inf.*, Vol. 19, No. 1, pp. 830-840, 2023.
[34] H. W. Haiyan Wang and H. M. Haiyan Wang, "A Lightweight V2R Authentication Protocol Based on PUF and Chebyshev Chaotic Map", *Journal of Computers*, Vol. 34, No. 2, pp. 99-112, 2023.
[35] X. Jin, N. Lin, Z. Li, W. Jiang, Y. Jia, and Q. Li, "A Lightweight Authentication Scheme for Power IoT Based on PUF and Chebyshev Chaotic Map", *IEEE Access*, Vol. 12, pp. 83692-83706, 2024.
[36] Q. Xie and Y. Yao, "PUF and Chaotic Map-Based Authentication Protocol for Underwater Acoustic Networks", *Applied Sciences*, Vol. 14, No. 13, p. 5400, 2024.
[37] A. Muñoz, A. Maña, and D. Serrano, "AVISPA in the Validation of Ambient Intelligence Scenarios", In: *Proc. of 2009 International Conf. on Availability, Reliability and Security*, Fukuoka, Japan, pp. 420-426, 2009.
[38] C. J. F. Cremers, "The Scyther Tool: Verification, Falsification, and Analysis of Security Protocols", *Computer Aided Verification*, pp. 414-418, 2008.