

A  
Project Report  
On  
**Data Freshness for Reinforcing Mutual Authentication in  
Wireless Body Area Network Applications**

Submitted to  
RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES  
RK VALLEY

in partial fulfilment of the requirements for the award of the Degree of  
**BACHELOR OF TECHNOLOGY**

IN  
**COMPUTER SCIENCE ENGINEERING**

Submitted by  
**S. Challa Rao (R170131)**  
**G. Venkata Sumanth (R171231)**

**P. Varsha (R171062)**

**T. Supriya (R170363)**

**P. Nikitha (R170283)**

Under the Guidance of  
**Mr. T SANDEEP KUMAR REDDY**

Assistant Professor



DEPARTMENT OF COMPUTER SCIENCE ENGINEERING  
RAJIV GANDHI UNIVERSITY OF KNOWLEDGE  
TECHNOLOGIES, RK VALLEY -516330

2019-2023

## DEPARTMENT OF COMPUTER SCIENCE ENGINEERING



### CERTIFICATE

This is to certify that the project report entitled **“Data Freshness for reinforcing Mutual Authentication in Wireless Body Area Network Applications”** a bonafide record of the project work done and submitted by S Challa Rao (R170131), G Venkata Sumanth (R171231), P Varsha (R171062), T Supriya (R170363), P Nikitha (R170283) for the partial fulfilment of the requirements for the award of B.Tech. Degree in Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies, RK Valley.

The report has been not submitted previously in part or full to this university or any other university or institution for the award of any degree or diploma.

#### INTERNAL GUIDE

T SANDEEP KUMAR REDDY  
Assistant Professor, Dept of CSE,  
RGUKT - RK VALLEY

#### HEAD OF THE DEPARTMENT

N SATYANANDARAM  
HOD CSE,  
RGUKT-RK VALLEY

# **DECLARATION**

We hereby declare that the project report entitled “A Systematic Review on Mutual Authentication in Wireless Area Body Network Applications” submitted to the Department of COMPUTER SCIENCE ENGINEERING in partial fulfilment of requirements for the award of the degree of BACHELOR OF TECHNOLOGY. This project is the result of our own effort and that it has not been submitted to any other University or Institution for the award of any degree or diploma other than specified above.

**S. Challa Rao (R170131)**

**G. Venkata Sumanth (R171231)**

**P. Varsha (R171062)**

**T. Supriya (R170363)**

**P. Nikitha (R170283)**

## ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and who's constant guidance and encouragement crown all the efforts success.

I would like to express my sincere gratitude to **Mr. T. Sandeep Kumar Reddy**, my project guide for valuable suggestions and keen interest throughout the progress of the project.

I'm grateful to **Mr. N. Satyanandaram , HOD CSE** , for providing excellent computing facilities and congenial atmosphere for progressing the project.

At the outset, I would like to thank **Honourable Director Madam, Mrs. K Sandhya Rani** , for providing all the necessary resources and support for the successful completion of my course work.

**S. Challa Rao (R170131)**

**G. Venkata Sumanth (R171231)**

**P. Varsha (R171062)**

**T. Supriya (R170363)**

**P. Nikitha (R170283)**

## ABSTRACT

WBAN deals with the sensitive physiological information collected by the biomedical sensors distributed over the human body, and these observations are exchanged to the healthcare provider through the Internet without affecting the user's comforts. Data freshness ensures the freshness of cryptographic parameters passed during the mutual authentication process of WBAN. Achieving data freshness includes various challenges in WBAN due to its resource constraints and limited applicability.

However, the wireless channels are prone to various threats due to their openness, and safeguarding such a hazardous and valuable environment is essential. We design a four-factor authentication protocol for resource mining. The formal verification of the proposed protocol is carried out using the Burrows–Abadi–Needham (BAN) logic and real-or-random (ROR) model. Its robustness is ensured by using the informal security analysis. The protocol is simulated using ProVerif to ensure the secrecy of variables and security from various attacks. Finally, a comparative analysis with the existing protocols is carried out to show its efficacy in a practical environment.

# TABLE OF CONTENTS

Abstract	v
Table of Contents	vi

Chapter No.	Description	Page No.
-------------	-------------	----------

## **PROJECT - 1**

1	Introduction	1
2	Challenges in Data Freshness	3
3	Session Key	4
4	Mechanisms for Data Freshness	5
5	Conclusion & Future Scope	8

## **PROJECT - 2**

1	Introduction	9
2	Wu et al.'s Protocol	10
3	Proposed Protocol	11
4	Security Analysis	14
5	Protocol Implementation-OCaml	16
6	Simulation Analysis	18
7	Source Code	20
8	Conclusion	24
9	References	25

## **PROJECT - 1**

### **1. INTRODUCTION**

In real time, the medical professional accesses the real time observation of the patient through the hub node or local server. Accessing the data over the insecure channel forces the security analyst to develop the better authentication protocol to identify the data authenticity of the information. Unless it will lead to various threats and vulnerabilities which results in misleading intervention of the medical professional. Data confidentiality and privacy are the prime requisites for data transmission in WBAN because it handles sensitive information concerning the health conditions of the patients. To ensure authenticity in resource constrained environment, many security analysts designed various authentication protocols to fulfill the security goals

A wireless body area network (WBAN) is a type of wireless network that is designed to allow communication between small, wearable devices that are placed on or inside the human body. These devices are commonly referred to as body sensors or biosensors, and they are typically used to monitor a person's health or fitness levels.

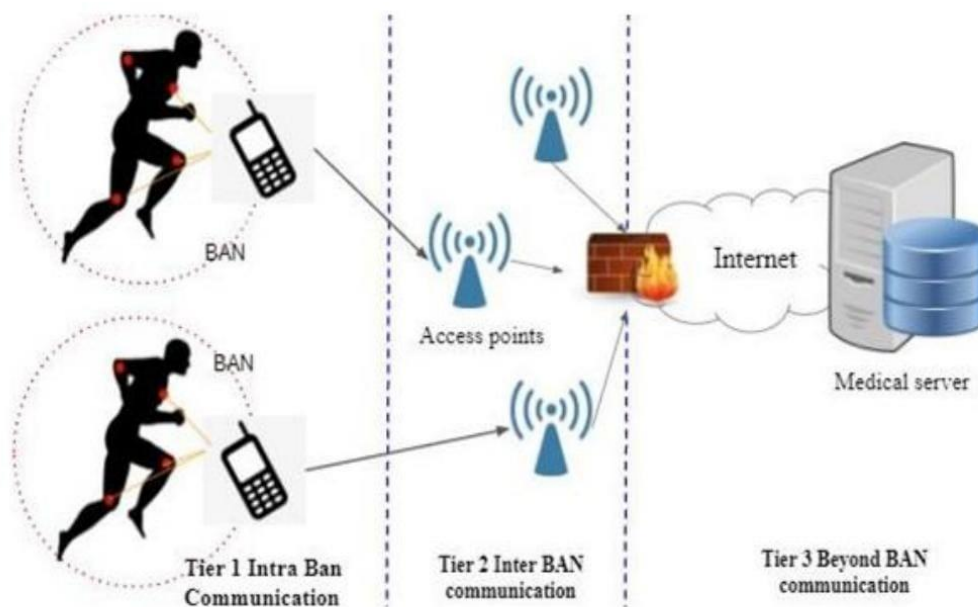
WBANs are becoming increasingly popular in the field of healthcare, as they offer a non-invasive way to monitor patients in real-time. For example, a patient with a heart condition may wear a biosensor that constantly monitors their heart rate and sends this information to their healthcare provider. The healthcare provider can then use this information to make informed decisions about the patient's treatment plan.

There are many challenges associated with designing and implementing a WBAN. One of the biggest challenges is ensuring that the biosensors are reliable and accurate, as even small errors in the data can have serious consequences. Another challenge is ensuring that the network is secure and that patient data is protected. Despite these challenges, WBANs have the potential to revolutionize the field of healthcare by providing a new level of real-time monitoring and personalized care.

## 1.1 WBAN COMMUNICATION ARCHITECTURE

The security issues in wireless body area networks, it might be significant to investigate the structure of the WBAN communication. The classification of tiers depends on whether the communication takes place between the internal entities of WBAN or outside the WBAN. Based on the location and features of the nodes, - Simple single hop network topology in which the sensor nodes and hub nodes are in the direct communication range, and no need to use the relay nodes. The latter is a centralized two-hop star topology in which the sensor and hub node is not in the direct communication range.

### ARCHITECTURE OF WBAN SYSTEM



**Fig. 1.** Architecture of a WBAN system

Above fig. illustrates the nodes disseminated by the specific distance in the network based on its applicability. The three types of nodes which are used in the WBAN are:

**Sensor nodes (End nodes):** Sensor nodes are the hardware peripheral units that are used to collect and exchange physiological information to the local server.

**Relay node (Intermediate nodes):** If the sensors and hub nodes are not in the proper network communication range, these are the intermediate nodes to provide enhanced coverage.

**Hub node (Local server):** These nodes exchange information between the sensor and the medical advisor.



**Tier I Intra-WBAN communication**

The communication among the end nodes and the interaction between the end nodes and relay nodes constitute the tier I communication that helps to transfer the observations to the next tier of communication.

**Tier II Inter WBAN communication**

The communication between the relay nodes and hub nodes comprises the tier II communication in WBAN

**Tier III beyond WBAN communication**

In this tier, the hub node transmits the collected observations about the patient to the medical advisor. Finally, the medical advisor analyses eHealth reports of the patient and alerts the patient in case of emergencies.

**2. CHALLENGES IN EXHIBITING THE DATA FRESHNESS**

The *general challenges* are WBAN must provide reliable communication with latency and jitter in an acceptable range. WBAN must support recoverable methodologies in case of any node failures. Finally, it must extend to the heterogeneous wireless environment, and the data rate should be from 10Kbps to 10Mbps.

The *security challenges* are security vulnerabilities and WBAN must support the authentication and encryption mechanisms. But substantiating the conventional cryptographic algorithm leads to higher memory and energy requirement, key size, and processing capability. These cryptographic primitives will not be applicable for resource-constrained WBAN. So, security challenges involve the storage requirements, computational overheads and communication costs of the security primitives.

**The Security Vulnerabilities are**

- Eavesdropping attack
- Replay attack
- Impersonation attack
- Key Compromisation attack
- Gateway bypassing attack
- Node Capture attack

### 3. SESSION KEY ESTABLISHMENT

The Security association is used to uniquely identify the sensor node and local server together. The security association is generated at the network layer by validating the authentication credentials of the user. The system administrator verifies the identity of each node before providing the master key for key establishment. System admin at the medical server-side shares the master key or pre-shared key among the sensor node and local server to establish a new session key for further communication.

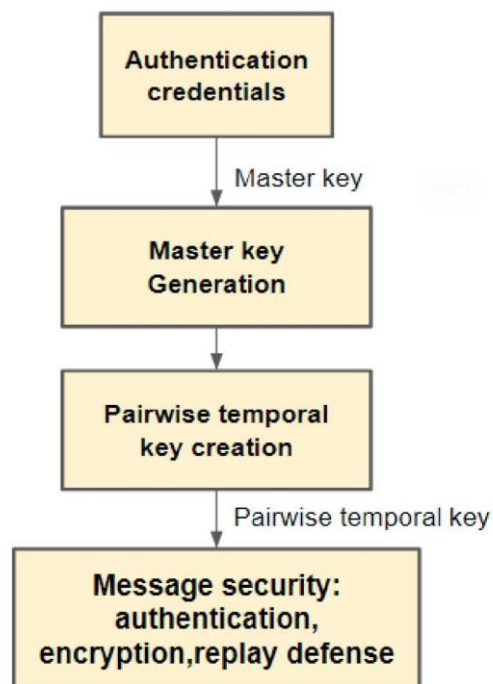
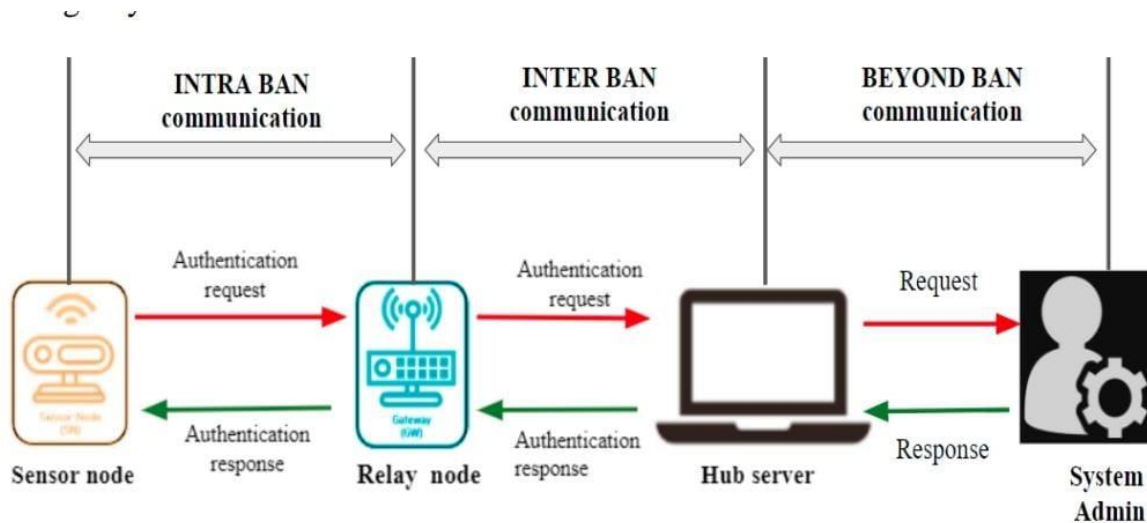


Fig. 2. Session key establishment in IEEE 802.15.6

Above figure explains the session key establishment as a result of the mutual authentication in Wireless Body area network. In this scheme, authentication credentials which may be static or dynamic attribute used to generate the master key from the system administrator in medical server. This master key provides the pairwise temporal key between the protocol entities and this pairwise temporal key is used for providing security properties in WBAN using the encryption, authentication and replay protection mechanisms. Among this process replay protection is established using the data freshness.

#### 4. MECHANISMS USED FOR DATA FRESHNESS IN WBAN

Data freshness is one of the security services which provide integrity and data protection for the cryptographic parameters passed during the mutual authentication protocols in WBAN. Data freshness is defined to verify whether the cryptographic parameters are current and not replayed from any other previous sessions.



**Fig. 3.** The workflow of WBAN

For providing high throughput and low latency in lightweight key agreement protocols, the sender has to send a message with some additional information to ensure the freshness of that message. In this case, one entity which is a system administrator needs to provide a key to all the other nodes such as sensor node, relay node, or hub node. Here nodes other than the system administrator do not have any part in session key generation. So, they investigate whether the session key distributed from the system administrator is a current or replayed one. For ensuring the freshness of the data, the protocol entities need to check some parametric values associated with the messages.

##### Parametric values for Data freshness

**Time Stamps:** The sender has to append the current timestamps with the message and send them to the receiver. This is used later for the security checking purpose

**Random Number:** It is chosen by the mobile node as an authentication request to the sensor node. And it is verified at the end of the process.

**Counters:** Whenever the message is sent by the sender, the sequence counter value must be appended with the message, and then the counter value is incremented by 1.

**Hybrid Methods:** By using the above three parametric values, some protocols use a hybrid mechanism for the data freshness. The sensor node needs to pick a random number (nonce), and the current timestamp concatenates it with the message to be transmitted. Upon validating the timestamp and the random number, the hub node accepts the message. In case, the smartcard user sends the message to the server with the current timestamp and the server checks its validity discussed in Algorithm 1 for verifying the cryptographic parameters.

---

**Algorithm 1 : verifying Data freshness of cryptographic parameters**

---

**At sensor**

- Step 1** Generate an identity of the node  $id_{sn}$ , a random number  $r_n$  and timestamp  $t_{sn}$  by the sensor node
- Step 2** Computes the cryptographic parameters and temporary id using  

$$tID = h(id_{sn} \oplus t_{sn} || r_n)$$
- Step 3** Send the cryptographic parameters  $\langle tID, y, A, B, t_{sn} \rangle$  to the hub node or local server.

**At Hub node**

- Step 4** Checks the validity of the timestamp  $t_{sn}$  with  $t_{hn}$
- Step 5** Derive the random number  $r_{new}$  based on the cryptographic parameters received
- Step 6** Calculate the temporary id  $tID_{new} = h(tID \oplus t_{sn} || r_{new})$
- Step 7** Checks the  $tID_{new} = tID$  to ensure the cryptographic parameters are current one and not replayed from the previous sessions.

## 4.1 LIGHTWEIGHT CRYPTOGRAPHIC PROTOCOLS

For securing the data communication, cryptographic mechanisms must indulge the end-to-end security among the protocol entities by excluding the interception of messages by unauthorized third parties. The communication cost and computational capability will be high to implement a better cryptographic mechanism for data transmission.

**Biometric based authentication schemes:** The dynamic biometric attribute is required for node authentication and transmission of data. If we use the static biometric attribute like face, iris and fingerprint, then it will not provide resistance over the stolen key problem, privileged insider attack and snooping attack. Here, the problem is the usage of the Error Correcting Codes increase the computational complexity and communication overhead with storage requirements.

**Block-chain based authentication schemes:** The current timestamps need to be appended with the identity of the node to ensure the freshness of the data. Although it verifies the freshness of

cryptographic parameters, these schemes do not provide much scalability in the dynamic node authentication phase. But these schemes need high throughput and low with negligible delay for each transaction carried in mutual authentication phase.

**Symmetric based schemes:** This consider the pre master key and the unique identifier of the nodes participating in the WBAN communication. For instance, the sensor node combines its unique identity with the current timestamp or random number to mask its identity using a temporary identifier for a particular session to ensure the properties like unlink ability and anonymity. To overcome this Password based Authentication and Key, two factor authentication, symmetric authentication scheme, novel authentication scheme and lightweight authentication protocol are proposed.

**Public key signature scheme:** the public key signature is formalized by the two non-identical keys. The main limitation of PKC is the higher communication and computation cost. For instance, RSA signature is based on the congruent theory which implies the data freshness indirectly in the authentication protocol.

**ID- based signature scheme:** In authentication, Trusted Third Party must generate the master time key and store it for validating the timestamps of the messages of the signer and verifier. Then TTP transmits the unique identifier for the hub node (signer) to establish the session by sending Identity of the signer, timestamp and signing key to the system administrator (verifier).

**Certificate less Signature schemes:** In this scheme timestamp is used to provide anonymity and unlink ability for the protocol entities. But for node authentication, timestamp value and identity of the node is easily traced out by an adversary. To eliminate the computational costs, non-pairing operations is utilized.

**Attribute based Cryptography:** It is using the set of context aware attributes in healthcare application using the access structure. Wu et al. proposed a role-based access control [30] for gaining the access of real time data observation of the patient. t. But these schemes do not have resistance over key escrow problem and depends on the single authority to be trusted.

**Physical unclonable functions (PUF):** The mutual authentication phase is set up through equipping the sensor nodes with PUF. Authentication takes place by sending the identity and random numbers of the sensor nodes to the sink node (local server). Here the replay attack is avoided. This is the typical example of random challenges which follows the request and response method to ensure the data freshness property.

## 5. CONCLUSION & FUTURE SCOPE

We analyzed the feasible solutions for avoiding the replay problem while establishing the session key between the sensor node and system administrator of the medical server in WBAN. In our analysis, we identified various methods used for data freshness maintenance in WBAN. Our results are summarized as follows:

Random challenges will be highly applicable when the quality of the random numbers is improved. Although each data freshness mechanism has some drawbacks, the hybrid mechanism is well suited for avoiding the replay attack by reducing the communication and computing cost in resource constrained environments. So, these hybrid methods would be a reliable solution for attaining end to end data freshness. But the timestamps play an effective role in asymmetric lightweight key agreement protocols and block-chain based key agreement protocols.

In real time there are various factors affecting the security and confidentiality of all the three-tier communication in WBAN. So, while designing the authentication and key establishment protocol, we must be careful in terms of the cryptographic primitives, computational cost and communication costs. In future work, data freshness in asymmetric cryptosystems such as Diffie Hellman key agreement, Bilinear mappings, ECC, PKI and chaotic encryption of medical images might be analyzed and compared in terms of delay, end to end reliability and throughput.

## **PROJECT - 2**

### **1. INTRODUCTION**

In the era of digital transformation, resource mining has become an integral part of numerous industries, ranging from cryptocurrencies to cloud computing. With the increasing reliance on online resources, the need for robust security measures to protect these valuable assets has become paramount. Authentication protocols play a crucial role in safeguarding sensitive information and ensuring that only authorized individuals can access and utilize these resources.

The combination of these four authentication factors forms a robust and multilayered security framework for resource mining operations. By incorporating both traditional and innovative authentication techniques, the protocol significantly enhances the integrity and confidentiality of sensitive resources. The four-factor authentication protocol minimizes the risk of unauthorized access, identity theft and resource misuse, thereby ensuring the safety and reliability of resource mining systems.

#### **1.1 THREAT MODEL**

Threat modeling involves identifying and communicating information about the threats that may impact a particular system or network. Security threat modeling enables an IT team to understand the nature of threats, as well as how they may impact the network.

The assumptions considered in our protocol to evaluate its security

1. An adversary **A** can overhear, morph, inject, replay, redirect, or delete the messages communicated over a public channel.
2. **A** cannot intercept the communications carried out over a private/secure channel.
3. **A** can pickpocket the smart device of a legal user of the system and obtains its secret credentials using the power analysis.
4. **A** can carry out an offline identity or password predicting attack separately. However, it cannot predict both parameters simultaneously in polynomial time.
5. **A** can obtain the data stored in the sensor's memory by physically capturing it and carrying out the power analysis.

## 2. WU ET AL.'S PROTOCOL

The protocol provides a variety of security functions, including PFS and resistance to privileged internal attacks, stolen smart card attacks, etc. We use the BAN (Burrows-Abadi Needham) logic, ProVerif tool and informal security analysis to prove the security. The symbols used in the protocol are given in Table

Symbols	Meaning
$SA$	System administrator
$U_i, S_j, GWN$	ith User, jth Sensor, Gateway
$ID_i, PW_i, B_i, SCN_i, PID_i$	ith user's identity, password, biometric, smart card number, psuedo-identity
$SID_j, ID_g, x$	jth sensor's identity, Identity of gateway, Gateway's secret key
$SK_u, SK_s, SK_g$	user's, senor's and gateway's Session Keys
$h(\cdot), \oplus, \parallel, BKG(\cdot)$	Hash function, xor function, Concatenation, Biometric key recovery function
$T_1, T_2, T_3, T_4$ & $\Delta T$	Timestamps & Transmission delay
$G_q, \mathbb{Z}_q^*$	Prime cyclic group, Multiplicative prime cyclic group

**Wu et al.'s Protocol suffers from following attacks:**

1. Privileged-Insider Attack
2. Man-in-the-Middle Attack
3. Sensor Node Capture Attack
4. Sensor Node Impersonation Attack
5. Gateway Impersonation Attack
6. Denial-of-Service Attack
7. Session Specific Transient Information Attack



### 3. Proposed Protocol

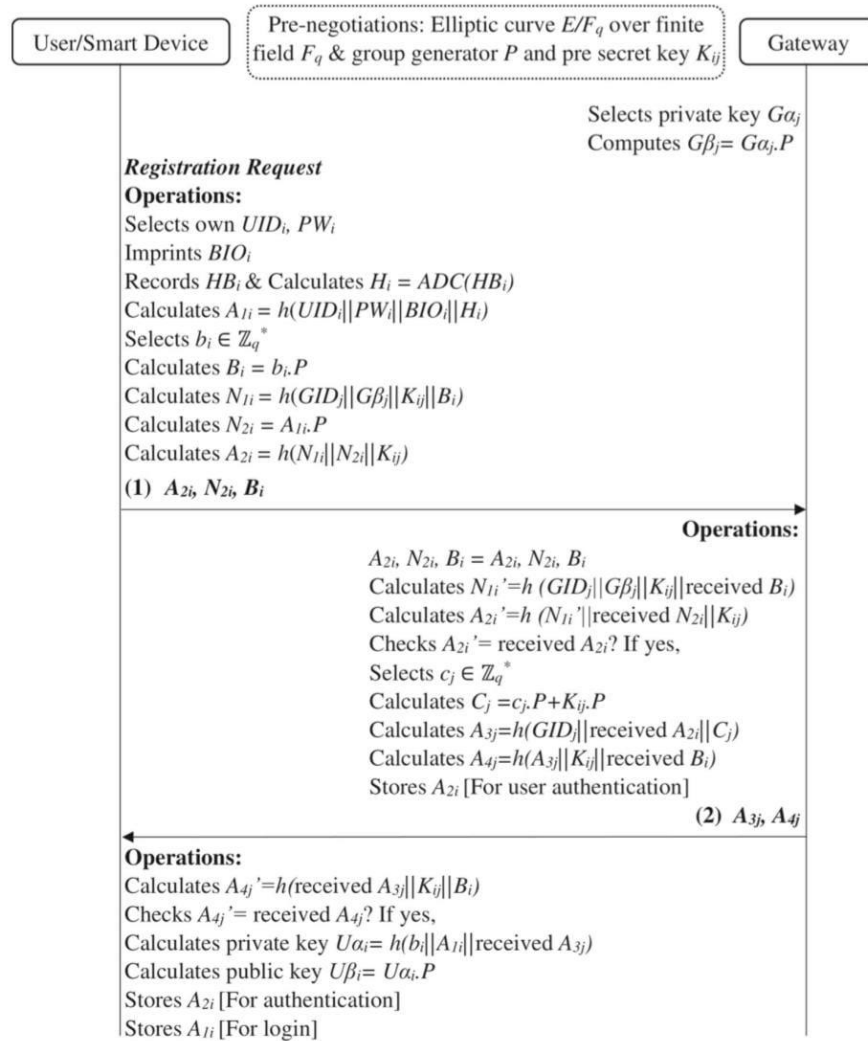
Our proposed protocol consists of five phases: initialization, user registration, sensor registration, user login, and mutual authentication. First, the system administrator (SA) initializes the parameters for all the communicating entities. Then the user and sensor nodes establish their public-private key pair after completing their respective registration phase. After becoming the legal part of the system, a user can log into the system to authenticate itself and access the sensor's data. The other communicating entities (gateway and sensor) get authenticated during the authentication process to ensure robust security. After successful authentication among all the communicating entities, the user and the sensor establish a secure session key for future data exchange

#### Initialization

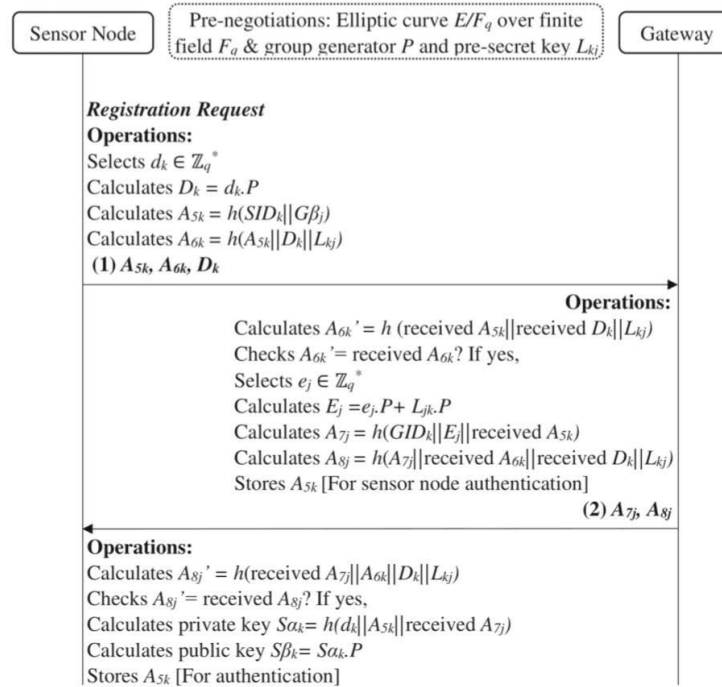
System administrator S A selects the following parameters for the participating entities  $U_i$ , GWN $_j$ , and SN $_k$  to initialize the system:

1. Identity SID $_k$ , GID $_j$  for sensor node and gateway, respectively.
2. Pre-secret key Lk $_j$  for sensor node and gateway, which is stored in their memories.
3. Pre-secret key Ki $_j$  for user and gateway, which is stored in the smart device and memory, respectively.
4. An asymmetric encryption and decryption algorithm.
5. A standard elliptic curve E over  $F_q$ .
6. A group generator P over  $E/F_q$  of order n. The gateway selects its private key G $\alpha_j$  to compute its public key G $\beta_j = G\alpha_j \cdot P$ .

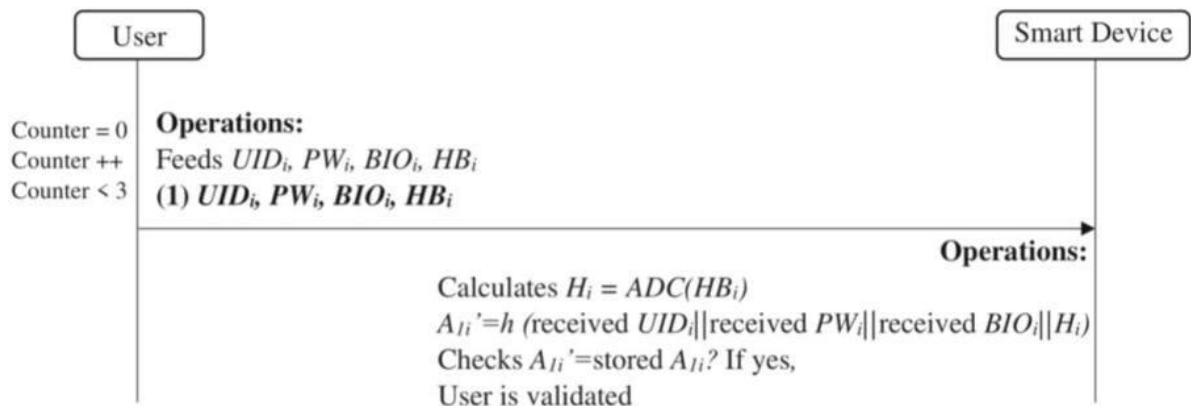
## User Registration



## Sensor Node Registration



## Remote User Login



## 4 Security Analysis of proposed protocol

The security analysis of the proposed protocol that includes security attributes, formal validation via BAN and by ROR model, and simulations using ProVerif.

### 4.1 Burrows–Abadi–Needham (BAN) Logic

BAN logic verifies the cryptographic protocols by using propositional logic. It is used to find the flaws in the security protocols, and its many variants have been developed over the years. The fundamental principle of the logic is the faith of an entity in the authenticity of the formula. It results in illuminating derivations which disclose precise errors in the protocol. The formal validation of our proposed protocol is shown by using the well-accepted BAN logic. The proof shows that the proposed protocol preserves the privacy of the shared session key only among the user  $U_i$  and sensor node  $SN_k$ . For validating the proposed protocol, we first idealize the message sent on a public channel, which is just the elaborate form of how each component of the message is composed. Then, few assumptions regarding random numbers, timestamps, shared keys, and variables are considered.

Rules	Explanations
Message Meaning Rule (MMR) If S trusts that L is shared with R and perceives $\langle I \rangle J$ , then S trusts, R formerly mentioned I	$\frac{S \models S \xleftrightarrow{L} R, S \triangleleft \langle I \rangle J}{S \models R \mid \sim I}$
Freshness Conjunction Rule (FCR) If S trusts, I is new, then S trusts newness of (I, J)	$\frac{S \models \#(I)}{S \models \#(I, J)}$
Nonce Verification Rule (NVR) If S trusts that I is new and R formerly mentioned I, then S trusts R which further trusts I	$\frac{S \models \#(I), S \models R \mid \sim I}{S \models R \mid I}$
Belief Rule (BR) If S trusts I and it also trusts J, then S trusts (I, J)	$\frac{S \models (I), S \models (J)}{S \models (I, J)}$
Jurisdiction Rule (JR) If S trusts, R has rights on I and R trusts I, then S trusts I	$\frac{S \models R \Rightarrow I, S \models R \mid \equiv I}{S \models I}$
Session Key Rule (SKR) If S trusts, I to be new and R trusts I, an essential factor of the session key, then S trusts that it shares the session key L with R	$\frac{S \models \#(I), S \models R \mid \equiv I}{S \models S \xleftrightarrow{L} R}$

## 4.2 Security Attacks

### Security Functionalities

The proposed protocol ensures all the security functionalities, whereas other protocols lack one or the other like anonymity, un-traceability, entity impersonation, offline password guessing, perfect mutual authentication, session key computation, man in the middle attack, session specific temporary information.

- Anonymity
- Untraceability
- Entity Impersonation
- Offline Password Guessing
- Perfect Mutual Authentication
- Perfect Password Secrecy
- Replay Attack
- Stolen Verifier
- Stolen Smart Card
- Denial of Services
- Sensor Node Capture
- Man in the Middle Attack
- Session Specify Temporary Information

## 5 Protocol Implementation

### 5.1 OCaml Language

OCaml stands for "Objective Caml". It is a variant of the ML programming language, and was originally developed by the French National Institute for Research in Computer Science and Control (INRIA) in the late 1990s.

OCaml is a statically typed, functional programming language that supports imperative, object-oriented, and concurrent programming paradigms.

OCaml is a compiled language, which means that you write code in OCaml, then you compile it to an executable binary that can run on your computer.

OCaml has a strong, static type system that allows for type inference, which means that the type of a variable can be inferred by the compiler without the programmer having to explicitly specify the type.

OCaml has a powerful module system that allows for abstraction and encapsulation of code.

OCaml has a syntax that is similar to that of other functional programming languages, such as Haskell and ML.

In OCaml, functions are first-class citizens, which means that they can be passed as arguments to other functions, returned as results from functions, and stored in data structures.

OCaml has built-in support for pattern matching, which is a powerful feature that allows you to match and destructure complex data structures such as lists, tuples, and records.

OCaml has garbage collection, which means that the programmer does not have to manage memory manually.

OCaml is used for a variety of applications, including scientific computing, system programming, web development, and more.

### 5.1.1 Structure of OCaml Language

Here are some common syntaxes in OCaml:

#### Variable declaration:

let variable\_name = value;;

Example: let x = 10;;

#### Function declaration:

let function\_name argument1 argument2 = function\_body;;

Example: let add x y = x + y;;

#### Conditionals:

if condition then true\_expression else false\_expression;;

Example: if x > 0 then "positive" else "negative";;

#### Loops:

There are no traditional for or while loops in OCaml. Instead, recursion is used to iterate over data structures.

#### Data types:

Integer: let x : int = 10;;

Float: let x : float = 3.14;;

Boolean: let x : bool = true;;

String: let x : string = "Hello, world!";;

List: let x : int list = [1; 2; 3];;

Tuple: let x : (int \* string) = (10, "hello");;

#### Pattern matching:

Used to match against different possible values of a variable.

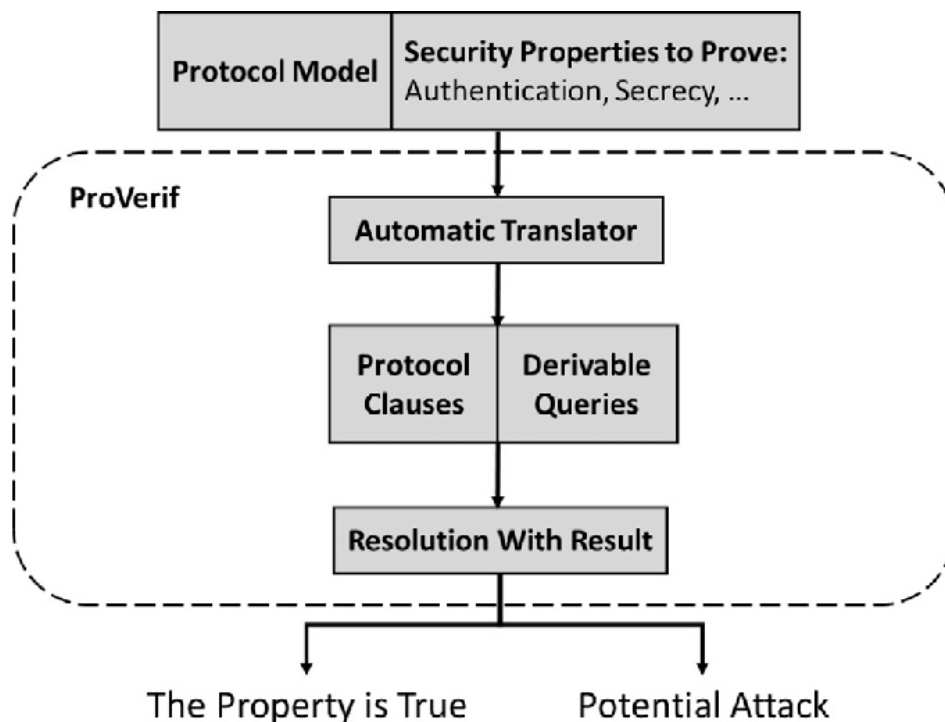
## 6.SIMULATION ANALYSIS OF PROPOSED PROTOCOL

### 6.1 Proverif Tool

**ProVerif** is a software tool for automated reasoning about the security properties found in cryptographic protocols. The tool has been developed by Bruno Blanchet.

Support is provided for cryptographic primitives including: symmetric & asymmetric cryptography; digital signatures; hash functions; bit-commitment; and signature proofs of knowledge. The tool is capable of evaluating reachability properties, correspondence assertions and observational equivalence. These reasoning capabilities are particularly useful to the computer security domain since they permit the analysis of secrecy and authentication properties. Emerging properties such as privacy, traceability and verifiability can also be considered. Protocol analysis is considered with respect to an unbounded number of sessions and an unbounded message space. The tool is capable of attack reconstruction: when a property cannot be proved, an execution trace which falsifies the desired property is constructed.

The ProVerif is a formal verification tool for simulating an authentication protocol. It uses the unbounded sessions and message space to verify the reachability of code, observational equivalence, and correspondence assertions. It uses a subset of the pi calculus and horn clause to model the protocol in the form of queries. It works on the well-known DY intruder model and allows a user to define various cryptographic operations like XOR, hash, encryption/decryption, etc. The correspondence relations among the communicating entities are checked via the events. In these processes, the step wise illustrations of the operations undertaken are coded where new gives the fresh variables, out () and in () mark the variables that navigate over the communication channels.





## 6.2 Performance Analysis

### Communication Cost

The identities, timestamps, and random numbers have been considered 32 bits long, SHA-1 hashed messages 160 bits long, and symmetrically encrypted messages 512 bits long. The proposed protocol has minimum total communication overhead in contrast to the protocols, making it useful in the bandwidth constraint environment; also, the lower communication overhead makes the transmission process more proficient.

User :  $5TH + 2Te + TA + Ta$

Gateway :  $8TH$

Sensor :  $4TH + 2Te + TA$

Total Computation Cost :  $17TH + 4Te + 2TA + Ta = 8.992 \text{ ms}$

### Computation Overhead

The protocol has lower computational complexity in comparison with the protocols. The protocol has lower complexity, but it suffers from various security breaches. Hence, our protocol has lower computational complexity while ensuring robust security, making it useful in practical scenarios where the lack of security could create havoc.

User (bits) : 224

Gateway (bits) : 448

Sensor (bits) : 224

$U_i + GWN_j + SN_k$  (bits) : 896

Storage (bits) : 320

Energy (mJ) : 4123.392

### Smart Device Storage Overhead

The proposed protocol incurs minimum storage overhead, making it memory efficient and ensuring the smooth functioning of the hardware devices. The negligible storage overhead aids in providing longer lifetime of the device.

## 7.SOURCE CODE

### ***\*\*Preliminaries\*\****

```

1 (*---channels---*)
2 free ch:channel.
3 free sch1:channel[private].
4 free sch2:channel[private].
5 (*Encryption and Decryption*)
6 type beta.
7 type alpha.
8 fun enc(bitstring,beta):bitstring.
9 fun dec(bitstring,alpha):bitstring.
10 equation forall m:bitstring,k1:beta,k2:alpha; dec(enc(m,k1),k2)=m.
11 (*---session keys---*)
12 free sku:bitstring[private].
13 free sks:bitstring[private].
14 (*---User's Secret Keys---*)
15 free Ualpha:bitstring[private].
16 free Ubeta:bitstring.
17 free K:bitstring[private].(*shared between user and gateway*)
18 free A2:bitstring[private].(*shared between user and gateway*)
19 (*---Gateway's Secret Keys---*)
20 free Galpha:alpha[private].
21 free Gbeta:beta.
22 free L:bitstring[private].(*shared between sensor node and gateways*)
23 (*---Sensor Node's Secret Keys 20---*)
24 free Salpha:bitstring[private].
25 free Sbeta:bitstring.
26 free A5:bitstring[private].(*shared between sensor node and gateway*)
27 (*---constants---*)
28 const P:bitstring.
29 free UIDi:bitstring[private].
30 free Pwi:bitstring[private].

31 free BIOi:bitstring[private].
32 free HBi:bitstring[private].
33 const SIDj:bitstring.
34 const GIDj:bitstring.
35 table user(bitstring).
36 table sensor(bitstring).
37 table gateway(bitstring).
38 (*---functions---*)
39 fun h(bitstring):bitstring.(*hash function*)
40 fun ecpm(bitstring,bitstring):bitstring. (*elliptic curve point multiplication*)
41 fun ecpa(bitstring,bitstring):bitstring. (*elliptic curve point multiplication*)
42 fun mul(bitstring,bitstring):bitstring. (*mathematical multiplication*)
43 fun add(bitstring,bitstring):bitstring. (*mathematical multiplication*)
44 fun con(bitstring,bitstring,bitstring):bitstring.(*string concatenation*)
45 fun con1(bitstring,bitstring):bitstring.(*string concatenation*)
46 fun con2(bitstring,beta,bitstring):bitstring.
47 fun con3(bitstring,beta):bitstring.
48 (*---queries---*)
49 query attacker(sku).
50 query attacker(sks).
51 query id:bitstring;inj-event(UserAuth(id))==>
52 inj-event(UserStart(id)).
53 (*---event---*)
54 event UserStart(bitstring).
55 event UserAuth(bitstring).
--

```

**\*\*Process User\*\***

```

56|(*--user process*)
57 let User=
58 new b:bitstring;
59 let A1 = h(con1(con(UIDi,PWi,BIOi),HBi)) in
60 let B = ecpm(b,P) in
61 let N1 = h(con1(con2(GIDj,Gbeta,K),B)) in
62 let N2 = ecpm(A1,P) in
63 let A2 = h(con(N1,N2,K)) in
64 out(sch1,(enc(A2,Gbeta),enc(N2,Gbeta),enc(B,Gbeta)));
65 in(sch1,(A3:bitstring,A4:bitstring));
66 let A4'=h(con(A3,B,K)) in (*72*)
67 if A4=A4' then
68 let Ualpha=h(con(b,A1,A3)) in
69 insert user(A2);
70 !
71 (
72 event UserStart (UIDi);
73 new T1:bitstring;
74 let I1=h(con(A2,K,Ubeta)) in
75 let I2=h(con(I1,T1,B)) in
76 out(ch,(I2,B,T1));
77 in(ch,(I8:bitstring,D:bitstring,T4:bitstring));
78 new T5:bitstring;
79 let I7'=h(con2(A2,Gbeta,K)) in
80 let I8'=h(con(I7',T4,D)) in
81 if I8'=I8 then
82 let SKu=ecpa(ecpm(b,Sbeta),ecpm(D,Ualpha)) in
83 0
84 ).

```

**\*\*Process Sensor\*\***

```

85|(*--sensor process*)
86 let Sensor=
87 new d:bitstring;
88 let D=ecpm(d,P) in
89 let A5=h(con3(SIDj,Gbeta)) in
90 let A6=h(con(A5,D,L)) in
91 out(sch2,(A5,A6,D));
92 in(sch2,(A7:bitstring,A8:bitstring));
93 let A8'=h(con1(con(A7,A6,D),L)) in
94 if A8'=A8 then
95 let Salpha=h(con(d,A5,A7)) in
96 let Sbeta=ecpm(Salpha,P) in
97 insert sensor(A5);
98 !
99 (
100 in(ch,(I4:bitstring,B:bitstring,T2:bitstring));
101 new T3:bitstring;(*110*)
102 let I3'=h(con2(A5,Gbeta,L)) in
103 let I4'=h(con(I3',T2,B)) in
104 if I4'=I4 then
105 let I5=h(con1(con(GIDj,L,Sbeta),I3')) in
106 let I6= h(con1(con(I5,T3,D),B)) in
107 out(ch,(I6,D,T3));
108 let SKs=ecpa(ecpm(d,Ubeta),ecpm(B,Salpha)) in
109 0
110 ).

```

**\*\*Process Gateway\*\***

```

111 |(*---Gateway's process---*)
112 let GWNReg1 =
113 in(sch1, (X:bitstring, Y:bitstring, Z:bitstring));
114 let x = dec(X, Galpha) in
115 let y = dec(Y, Galpha) in
116 let z = dec(Z, Galpha) in
117 let N1' = h(con1(con2(GIDj, Gbeta, K), z)) in
118 let A2' = h(con(N1', y, K)) in
119 if A2'=x then
120 new c:bitstring;
121 let C = ecpa(ecpm(c, P), ecpm(K, P)) in
122 let A3 = h(con(GIDj, C, A2')) in
123 let A4 = h(con(A3, z, K)) in
124 insert gateway(A2');
125 out(sch1, (A3, A4)).
126 let GWNReg2 =
127 in(sch2, (A6:bitstring, A5:bitstring, D:bitstring));
128 let A6' = h(con(A5, D, L)) in
129 if A6' = A6 then
130 new e:bitstring;
131 let E=ecpa(ecpm(e, P), ecpm(L, P)) in
132 let A7=h(con(GIDj, E, A5)) in
133 let A8=h(con1(con(A7, A6, D), L)) in
134 insert gateway(A5);
135 out(sch2, (A7, A8)).
136

```

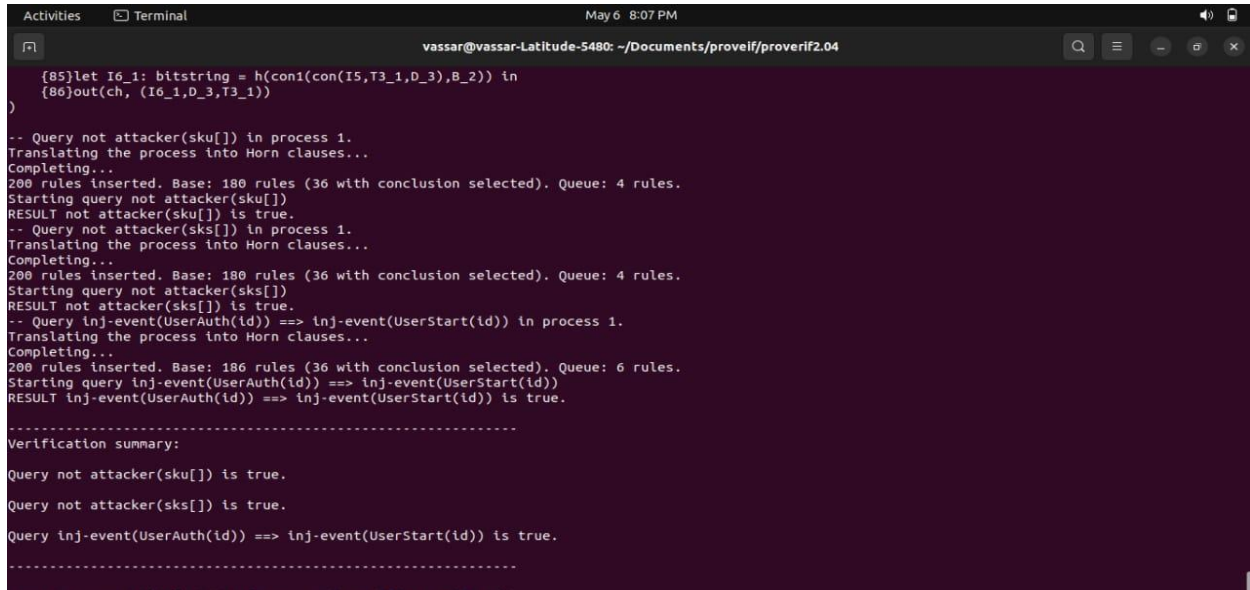
**\*\*Process Authentication\*\***

```

137 let GWNAuth =
138 in(ch, (I2:bitstring, B:bitstring, T1:bitstring));
139 new T2:bitstring; (*148*)
140 let I1'=h(con(A2, K, Ubeta)) in
141 let I2'=h(con(I1', T1, B)) in
142 if I2'=I2 then
143 event UserAuth(UIDi);
144 let I3=h(con2(A5, Gbeta, L)) in
145 let I4=h(con(I3, T2, B)) in
146 out(ch, (I4, B, T2));
147 in(ch, (I6:bitstring, D:bitstring, T3:bitstring));
148 new T4:bitstring;
149 let I5'=h(con1(con(GIDj, L, Sbeta), I3)) in
150 let I6'=h(con1(con(I5', T3, D), B)) in
151 if I6'=I6 then
152 let I7=h(con2(A2, Gbeta, K)) in
153 let I8=h(con(I7, T4, D)) in
154 out(ch, (I8, D, T4)).
155 let GWN = GWNReg1|GWNReg2|GWNAuth.
156 process!User|!GWN|!Sensor

```

## RESULTS



```

Activities Terminal May 6 8:07 PM
vassar@vassar-Latitude-5480: ~/Documents/proveif/proverif2.04

{85}let I6_1: bitstring = h(con1(con(I5,T3_1,D_3),B_2)) in
{86}out(ch, (I6_1,D_3,T3_1))
)

-- Query not attacker(sku[]) in process 1.
Translating the process into Horn clauses...
Completing...
200 rules inserted. Base: 180 rules (36 with conclusion selected). Queue: 4 rules.
Starting query not attacker(sku[])
RESULT not attacker(sku[]) is true.
-- Query not attacker(sks[]) in process 1.
Translating the process into Horn clauses...
Completing...
200 rules inserted. Base: 180 rules (36 with conclusion selected). Queue: 4 rules.
Starting query not attacker(sks[])
RESULT not attacker(sks[]) is true.
-- Query inj-event(UserAuth(id)) ==> inj-event(UserStart(id)) in process 1.
Translating the process into Horn clauses...
Completing...
200 rules inserted. Base: 186 rules (36 with conclusion selected). Queue: 6 rules.
Starting query inj-event(UserAuth(id)) ==> inj-event(UserStart(id))
RESULT inj-event(UserAuth(id)) ==> inj-event(UserStart(id)) is true.

-----
Verification summary:

Query not attacker(sku[]) is true.

Query not attacker(sks[]) is true.

Query inj-event(UserAuth(id)) ==> inj-event(UserStart(id)) is true.

-----

```

### Verification Summary:

Query not attacker(sku[]) is true

Query not attacker(sks[]) is true

Query inj-event(UserAuth(id)) ==> inj-event(UserStart(id)) is true

## 8. CONCLUSION

A number of attacks, such as privileged-insider, man-in-the-middle, gateway impersonation, sensor node impersonation, denial-of-service, session specific transient information, and sensor node capture, affect the Wu et al. protocol. Additionally, it has key leakage, adds to communication overhead, and fails to impart sufficient confidentiality and anonymity. We have talked about a resource mining four-factor authentication mechanism. Using the BAN logic, and ProVerif simulation, we have verified the security strength of the suggested protocol. Comparable protocols fall short of the proposed protocol's security features. The suggested protocol demonstrates that it satisfies the criteria for energy efficiency and that it is appropriate for any real-world resource-constrained context by achieving required security characteristics with lower overheads. In threshold attribute-based encryption and ciphertext, the key escrow issue was addressed.

## 9. REFERENCES

- Arathi Paper
- Two Factor Authentication 1
- Two Factor Authentication 2
- Base Paper to Four Factor
- Rangwani Four Factor
- Wu, F.; Li, X.; Xu, L.; Vijayakumar, P.; Kumar, N.: A novel three factor authentication protocol for wireless sensor networks with IoT notion. IEEE Syst. J. 15, 1120–1129 (2020)