Abstract—Recently, Kumar and Chand proposed an anonymous authentication protocol for wireless body area network. They claimed that their scheme meets major security requirements and is able to resist known attacks. However, in this paper we demonstrate that their scheme is prone to traceability attack. Followed by this attack, an attacker can launch a man-in-the-middle attack and share a session key with the victim node, and hence the scheme does not achieve secure authentication. Also, we show that this protocol does not provide perfect forward secrecy which considered as a key security property in the design of any secure key agreement protocol.

Index Terms—Wireless body area network (WBAN), perfect forward secrecy, traceability, man-in-the-middle (MITM), authentication

I. INTRODUCTION

THE recent advancements in wireless technology, embedded systems, integrated circuits, wearable devices and nano-technology have led to emergence of a new paradigm in Internet of Things (IoT)-based medical system called Wireless Body Area Networks (WBANs). In a WBAN system, micro/nano sensors are placed in or around human body that collect biological information of a patient and send the collected information involving body temperature, heart rate, blood pressure, etc. to a medical system through wireless communication [1].

WBAN technology has a vast range of medical and nonmedical applications that the common goal of all of them is continuous monitoring of a target body [2]. In medical science, multiple low-power sensors implanted in different nodes measure patient’s physiological data and send them via a wireless channel to a powerful cloud server for further process. Since the gathered data is the personal attributes of a human body, so the secrecy and privacy of the data should be preserved. This necessitates to make the communication between nodes and servers secure in a WBAN ecosystem.

Although, IEEE 802.15.6 standard has proposed security rules and mechanisms for in/on the body sensor acting in a WBAN communications [3], but the security of its presented protocols has been severely challenged by the researchers and it seems to be far from being accepted in its current form [5], [6].

The main security challenge in WBAN pertains to design secure key agreement protocol. In recent years, numerous key agreement protocols have been designed by the scientists [7]-[12] that either try to improve the security issues of the IEEE 802.15.6 standard, or offer a new scheme suitable for secure WBAN communications. The security analysis of a large number of these schemes are brought in [4].

Recently, an identity-based anonymous authentication and key agreement (IBAAKA) protocol has been presented by Kumar and Chand [13]. Although the scheme provides some major security properties like session key security, anonymity, resilience to many known attacks, etc. coupled with desirable communication and computation costs and also claimed that guarantees mutual authentication, perfect forward secrecy and user revocation, but in this paper we show that it fails to traceability attack followed by man-in-the-middle (MITM) attack that totally breaks mutual authentication. We also observed that their scheme is unable to achieve perfect forward secrecy, one of the main security property a secure key agreement protocol must provide. Besides this weaknesses, it is found that no user revocation facility has been proposed by the authors. In this paper, we cryptanalyzed this scheme and describe the aforementioned attacks in details. We briefly suggest some improvement techniques for future design to prevent these types of attacks.

II. REVIEW OF KUMAR AND CHAND’S SCHEME

The notations used in Kumar and Chand’s scheme are listed in Table I. Generally, there are three entities participating in the scheme with following roles: 1) $NM$: a trusted third party that generates and publishes security parameters to $LN$ and $CS$ through a registration mechanism. 2) $LN$: A resource-constraint sensor that registers with $NM$ and get security parameters allows it to authenticate with a legitimate cloud server. 3) $CS$: A cloud server that registers with $NM$ to be able to establish authentication session with a legal leaf node $LN$. It possesses a high computational power and a large amount of storage space.

Kumar and Chand’s scheme consists of three phases as follows:

1) Setup phase: $NM$ chooses an $l$-bit random number $q$, a
multiplicative cyclic group \(G_2\) and an additive cyclic group \(G_1\) both of order of \(q\). Let \(P\) be the generator of \(G_1\). \(NM\) also selects five secure hash functions \(H_1: \{0,1\}^* \rightarrow \mathbb{Z}_q^*, H_2: \{0,1\}^* \times G_1 \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*, H_3: G_1 \rightarrow \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*, H_4: \{0,1\}^* \times \{0,1\}^* \times G_1 \rightarrow \mathbb{Z}_q^*, H_5: \{0,1\}^* \times \{0,1\}^* \times G_1 \times G_1 \rightarrow \mathbb{Z}_q^*\). \(NM\) chooses \(s_0 \in \mathbb{Z}_q^*\) as its master key and computes its public key \(P_0 = s_0P\). Finally, \(NM\) publishes parameters \((l, q, P, P_0, G_1, H_1, H_2, H_3, H_4, H_5)\).

2) **Registration:** \(NM\) registers \(CS\) and \(LN\) as follows:
   a. **LN Registration:** \(LN\) chooses a random identity \(ID_{LN} \in \{0,1\}^*\) and sends this value and the registration request to \(NM\). \(NM\) first checks the availability of this request. Then it selects a random number \(r_{LN} \in \mathbb{Z}_q^*\) and computes \(R_{LN} = r_{LN}P\). It sends \((s_{LN}, R_{LN})\) to \(LN\) as its private key.
   b. **CS Registration:** In the similar way, \(CS\) gets its private key \((s_{CS}, ID_{CS})\) from \(NM\).

3) **Authentication:** The authentication phase has been shown in Fig. 1. In this phase \(LN\) and \(CS\) authenticate each other in following steps.
   a. \(LN\) selects a random number \(x \in \mathbb{Z}_q^*\) and computes \(X = xp\). \(g_{LN} = x(\text{id}_{CS} + H_1(\text{id}_{CS} || R_{CS})P_0)\). Then it calculates \(W = H_2(g_{LN}) \oplus (ID_{CS} || R_{LN} || T_{CS} || y || X)\) where \(T_{CS}\) is the current timestamp of \(CS\) and \(k = s_{LN} + xH_2(ID_{LN} || R_{LN} || T_{CS} || X)\). It sends login message \((W, X)\) to \(CS\).
   b. Upon receiving \((W, X)\), \(CS\) computes \(g_{CS} = g^{s_{CS}X}\) and extract \(ID_{LN} - R_{LN}\) from \((ID_{LN} || R_{LN} || T_{CS} || y || X)\) where \(h = H_2(ID_{LN} || R_{LN} || T_{CS} || X)\). The equality holds as below.

   \[
   kp = (s_{LN} + xh)P = (r_{LN} + s_0H_1(ID_{LN} || R_{LN}) + xh)P = r_{LN} + H_1(ID_{LN} || R_{LN})P_0 + hX
   \]

   If above equation holds, \(CS\) picks a random element \(y \in \mathbb{Z}_q^*\) and a fresh timestamp \(T_{CS}\) and sets \(Y = yp\), \(t = H_4(T_{CS} || ID_{CS} || ID_{CS} || Y || X)\). After computing values \(key_{CS} = ytX, SK_{CS} = H_5(key_{CS} || ID_{LN} || ID_{CS} || Y || X)\), it sends message \((t, T_{CS}, Y)\) to \(LN\) and stores \(SK_{CS}\) as a shared session key.

   c. When \(LN\) receives message \((t, T_{CS}, Y)\), it checks the timestamp freshness. If it was fresh, \(LN\) verifies whether \(t = H_4(T_{CS} || ID_{CS} || ID_{CS} || Y || X)\) holds. If not it aborts the session. Otherwise, it sets \(key_{CS} = xty\) and computes the shared session key with \(CS\) by \(SK_{LN} = H_5(key_{CS} || ID_{LN} || ID_{CS} || Y || X)\).

### III. Cryptanalysis of Kumar and Chanda’s scheme

In this section, we show that how traceability and MITM attacks can be applied on Kumar and Chanda’s Scheme and as a result it cannot achieve secure authentication. Moreover, we demonstrate that their scheme does not provide perfect forward secrecy which is an essential security property in an authenticated key agreement protocol.

#### A. Traceability Attack

The proposed protocol by Kumar and Chanda is subject to traceability attack in one of the following scenarios.

1) \(A\) be an external adversary who has obtained leaf node’s identity \(ID_{LN}\).

2) \(A\) be a cloud server who has established an authentication session with \(LN\) at least once.

We show that in both above scenarios, \(A\) can trace \(LN\) in polynomial time. Without loss of generality, we only discuss the second scenario and the veracity of the first one can be tested in similar way. Let \(CS_j\) establishes an authentication session with \(LN\). So, it knows \(ID_{LN}\). Also, we assume \(CS_j\) has the full control of the transmitted messages in the channel. We show that \(CS_j\) can easily trace any session initiated by \(LN\).

Below steps describe the details of the traceability attack.

**Step 1:** \(LN\) sends login message \((W, X)\) to \(CS_j\) where \(j \neq i\).

**Step 2:** In response, \(CS_j\) authenticates \(LN\) and sends login response \((t, T_{CS_j}, Y)\) towards \(LN\).

**Step 3:** \(CS_j\) intercepts \((t, T_{CS_j}, Y)\) and checks the correctness of \(t = H_4(T_{CS_j} || ID_{CS_j} || ID_{CS_j} || Y || X)\). If equation holds, it means that \(LN\) is successfully traced.

#### B. Man-In-The-Middle Attack

Let \(A\) be the adversary who has performed a successful traceability attack as described in section III-A. Similar to section III-A, we only investigate the case that \(A\) be a cloud...
server adversary like $CS_I$ who has established an authentication session with leaf node $LN$. The MITM attack procedure is as follows, as is also shown in Fig. 2.

**Step 1:** Let $LN$ be traced successfully by $CS_I$ as described in section III-A. Now, $CS_I$ selects a random element $y' \in Z_q^*$, sets $Y' = y'P$ and picks a new timestamp $T_{CS_I}$. Then instead of the message $(t, T_{CS}, Y)$, it sends $(t', T_{CS}, Y')$ to $LN$ where $t' = H_4(T_{CS}, |ID_{LN}| |ID_{CS}| |Y'| |X|$ and $X$ is an intercepted value from login message. Further, $CS_I$ computes the shared session key of impersonated $CS_I$ with $LN$ as $SK_{imp-CS_I} = H_5(y't'X | ID_{LN}| |ID_{CS}| |Y'| |X)$.  

**Step 2:** $LN$ receives $(t', T_{CS}, Y')$ and verifies $t' = H_4(T_{CS}, |ID_{LN}| |ID_{CS}| |Y'| |X)$. Now, it sets the shared session key with impersonated $CS_I$ as $SK_{LN} = H_5(x't'Y | ID_{LN}| |ID_{CS}| |Y'| |X)$.  

It is obvious that $CS_I$ is authenticated for $LN$ and both shared the same session key. From now on, $CS_I$ can securely communicate with $LN$ on behalf of $CS_I$. This is a crucial security weakness since by launching this MITM attack, cloud servers can impersonate each other to victim leaf nodes and access sensitive and unauthorized data.

### C. No Perfect Forward Secrecy

A security protocol provides perfect forward secrecy if an adversary $A$ obtains both secret keys of a leaf node $LN$ and a cloud server $CS$, he cannot get the session key of previous established sessions. Kumar and Chand claimed that their scheme has perfect forward secrecy property. However, we demonstrate that it cannot maintain this security requirement. Let private keys of both parties means $s_{CS}$ and $s_{LN}$ be revealed to an adversary $A$. Also, we assume $A$ has intercepted all transmitted messages in the channel. Since $A$ knows $s_{CS}$, he can calculate $g_{CS} = s_{CS}X$. Then he extracts $ID_{LN}, R_{LN}, T_{LN}, k$ from $(|ID_{LN}| |R_{LN}| |T_{LN}| |k|) = H_2(g_{CS}) \oplus W$. Now, $A$ can form the following equation

$$xH_2(|ID_{LN}| |R_{LN}| |T_{LN}| |X|) = (k - s_{LN}) \tag{1}$$

Using the Euclidean algorithm, $A$ can properly find $H_2(|ID_{LN}| |R_{LN}| |T_{LN}| |X|)^{-1}$ from (1) and then compute $x = H_2(|ID_{LN}| |R_{LN}| |T_{LN}| |X|)^{-1} (k - s_{LN})$. He also can find $key_{LN} = xtY$ and $SK_{LN} = H_5(key_{LN} | ID_{LN}| |ID_{CS}| |Y'| |X)$ accordingly. $A$ has obtained the session key that means the scheme does not provide perfect forward secrecy.

### D. No User Revocation Procedure

Kumar and Chand claimed that their scheme supports user revocation. But in fact, no revocation procedure has been proposed by the authors and the scheme lacks user revocation mechanism.

### IV. SEVERAL SUGGESTIONS FOR IMPROVEMENT

In section III, we have described security weaknesses of Kumar and Chand’s scheme. Here we suggest several countermeasures could be applied to the protocol enable it to withstand the mentioned attacks.

1) The main reason the protocol is traceable is that in the authenticator $t = H_4(T_{CS}, |ID_{LN}| |ID_{CS}| |Y'| |X)$, the only anonymous value to an adversary is the leaf node’s identity $ID_{LN}$ and if somehow he gets it, he can trace the leaf node. As a direct way of improvement, one can add a session-dependent value like $g_{CS}$ in $H_4$ hash function of $t$ which makes it meaningful only for those who actually participate in the authentication session.

2) The MITM attack has shown in section III-B is feasible if an adversary can trace the leaf node. So securing the protocol against the traceability attack will prevents the MITM attack too.

3) To the best of our knowledge, there are no straightforward techniques can use to achieve perfect forward secrecy property. But, it is obvious that any improvement should be in this the way that preserves the security of the ephemeral secret $x$ in case of revealing long term keys.

### V. CONCLUSION

In this paper, we have cryptanalyzed the most recently authentication scheme for WBAN by Kumar and Chand. Although authors claimed that their scheme has a high level of security comparing with the prior proposed schemes, but we have shown that it suffers from traceability attack followed by MITM attack which allows cloud servers to impersonate each
other to a common leaf node. Moreover, we have pointed out that unlike the author claims, it fails to provide perfect forward secrecy and also user revocation facility. We have suggested several countermeasures to prevent aforementioned security weaknesses. Further work should concentrate on improving Kumar and Chand’s scheme or designing a new anonymous authenticated key agreement protocol for secure WBAN.

REFERENCES


