Received: September 22, 2018

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Secure Model for Machine to Machine Device Domain Based Group in a Smart City Architecture

Mariya Ouaissa[1]* **Abdallah Rhattoy[2]**

[1] *Information and Communication Systems Engineering Research Group, High School of Technology, Mathematical Modeling and Computer Science Laboratory, Ecole Nationale Supérieure des Arts et Métiers Moulay-Ismail University, Meknes, Morocco*
[2] *Department of Computer, Information and Communication Systems Engineering Research Group, High School of Technology, Moulay-Ismail University, Meknes, Morocco*
* Corresponding author's Email: mariya.ouaissa@edu.umi.ac.ma

Abstract: The new Internet of Things (IoT) applications are enabling Smart City initiatives worldwide. It provides the ability to remotely monitor, manage and control devices. The main features of a smart city include privacy of users and security of communication. The distributed architecture of IoT aims at providing location awareness and low-latency interactions to Machine-to-Machine (M2M) applications. In this context, the Long Term Evolution (LTE) technology and its evolutions are expected to play a major role as a communication infrastructure that guarantees low deployment costs and embedded security, we show how a network deployment that exploits Device-to-Device (D2D) communications and Wireless Sensor Network (WSN), currently under definition within 3rd Generation Partnership Project (3GPP) and non-3GPP access network, can be employed to support efficient communication between M2M devices (nodes and smart objects) and the core network in LTE system. In this paper, we propose a secure group model include different authentication mechanisms combined between D2D and WSN technologies that allow the protection of M2M devices through strong encryption and authentication means, so that devices can benefit from high security functionalities without however having to execute computationally intensive operations with a low communication complexity. The results obtained show that our model firstly solve the initial key establishment and integrity problems in the presence of the inside adversaries in networks and offers best performance than the standard Bluetooth in D2D link. Secondly our proposed examine the area of authentication for sensor networks by make a combination between algorithms such as TinyPK and Timed Efficient Stream Loss-Tolerant Authentication (TESLA) in order to secure the link between sensors and achieve a high levels of security.

Keywords: IoT, M2M, MTC, LTE, 3GPP, Security, Group authentication, D2D, WSN.

1. Introduction

Internet of Things (IoT) [1] is a novel paradigm that is shaping the evolution of internet. It envisions billions of physical objects or things such as sensors, automobiles, refrigerators, thermostats, industrial robots, tablets, smartphones, etc. that could be connected anytime and anywhere to the internet through specific protocols for information exchange and communications, in order to achieve intelligent recognition, location, tracking, monitoring and management [2]. Things or objects need identifying, sensing, networking and processing capabilities to make the IoT paradigm a reality. Recent advances in the fields of wireless technology, advanced communications and intelligent systems have exhibited a strong potential and tendency on improving human life in every fact. We can realize the IoT for smart city [3, 4] through super computers, cloud computing, cyber-physical and Machine to Machine (M2M) technologies. M2M is considered an integral part of the Internet of Things movement, and the term is widely used by industry experts. When a machine communicates with another machine to accumulate information and exchange data, then it is called M2M communication [5]. It may use various wireless and wired networking. In recent days, M2M communication includes the transmission of data to personal appliances. Cellular wireless technologies have been considered the most promising candidates to support M2M communication for its ubiquitous coverage, good support of user mobility, high data rates and flexible spectrum usage. Consequently, the 3rd Generation Partnership Project (3GPP) has standardized M2M as Machine Type Communication (MTC) in Long Term Evolution (LTE) and its advancements (LTE-A) [6].

M2M is expected to offer advanced connectivity of devices, systems, services and covers a variety of protocols, domains, and applications. The interconnection of these embedded devices including smart objects can be based on group communication for a set of M2M devices that have certain common principles (e.g., belong to the same application, within the same region, etc.). The M2M device domain can support recent technologies in order to build heterogeneous devices. Among those technologies, there are two that fall into the scope of this work: Device-to-Device (D2D) communication and Wireless Sensor Network (WSN).

Along with the MTC, 3GPP has introduced a new technology called D2D communications. D2D refers to the ability of one wireless device to communicate, via direct link, with another using the spectrum that is available for regular cellular communications. It is noteworthy that existing technologies, such as Wi-Fi and Bluetooth, already allow devices to directly communicate with each other. On the other hand, WSN are frequently used for data gathering applications, such as military sensing and tracking, environment monitoring, patient monitoring, etc. The WSN require the system to be smart and adaptive to the changes in the application environment, task objectives, and topological variance, among others. There are a growing number of applications using WSN for smart communications, and the last few years have witnessed the development of many innovative solutions for their commercialization and standardization.

M2M that will bring a revolution in different sectors for example: emergency services, marketing, security, etc. Consequently, integrity and confidentiality of transmitted data as well as the authentication of the services offering that data is crucial. Hence, security is a critical functionality for the M2M in cellular network. Current D2D protocols cannot guarantee confidentiality or integrity of communications since malicious intermediate nodes can perform Main-In-The-Middle (MITM) or replay attack during the transmission. In addition, security is an important issue in sensor network. Due to the nature of communication and the kind of data they are going to handle, it is important to have the capability in the network to establish trusted communication.

In this paper we follow an alternative research direction that is based on two main components: the use of D2D communications like a link to secure group communication of MTC Devices (MTCD) and the use of WSN to connect a group of sensor in a secure way to convey the data in an efficient way over both 3GPP and non-3GPP access network. We focus on a typical reference M2M environment where different types of devices produce small data to be uploaded to the network over the LTE infrastructure. Our goal in this work is to propose two approaches for secure group communication. The first is for D2D links, in this approach, we use the framework of Bluetooth protocol based on Ciphertext-Policy Attribute-Based Encryption (CP-ABE) to solve the problems of initial key establishment and integrity problems with regard to sharing the initial secret information safely under the attacks. For the second approach we examine the area of authentication for sensor networks by make a combination between algorithms which could be used. This combination is recommended because TinyPK is able to enable the communicating nodes by creating a secret shared key. In order to secure the link between sensors it is recommended that Timed Efficient Stream Loss-Tolerant Authentication (TESLA) is used as it allows the key exchange is verified and can achieve a high levels of security.

The rest of the paper is organized as follows: in section 2 we describe the network architecture of a smart city, we present the description of a general security architecture along with its basic procedures and we explain several new security issues and existing solutions of M2M communication. In the third section, we discuss our method based on group combined between two approaches. The first is for D2D links, in this approach, we verify the solution by the security protocol verification tool, Automated Validation of Internet Security Protocols and Applications (AVISPA), in order to ensure his level of security and we evaluate some performances in term of communication cost and computation cost in section 4. The second approach for wireless sensor network, we give a comparison study between some existing mechanisms to reach the optimal one in order to secure the group communication in WSN in section 5. Finally, we draw our conclusions and future work in section 6.

2. System model

In this section, we introduce the network architecture, and we present some security threats and corresponding solutions according to 3GPP.

2.1 Network architecture for M2M

Key network elements for M2M communications, consisting of the following parts (Fig. 1) [7]:

- **M2M Area Network (Device Domain):** A device capable of replying to request for data contained within those device or capable of transmitting data contained within those devices autonomously. This domain provide connectivity between M2M devices and M2M gateways (e.g. personal area network).
- **M2M Gateway:** Use M2M capabilities to ensure M2M devices inter-working and interconnection to the communication network.
- **M2M Communication Networks (Network Domain):** Communications between the M2M gateway(s) and M2M application (e.g. xDSL, LTE, WiMAX, and WLAN).
- **M2M Applications:** Contains the middleware layer where data goes through various application services and is used by the specific business-processing engines.

2.2 Security architecture for M2M

Security architecture for M2M can be divided to three different areas, as shown in Fig. 2 [8].

(A)Security for M2M communication between the MTC device and 3GPP network can be further divided to:

(A1) Security between the M2M device and the core network through 3GPP access network.

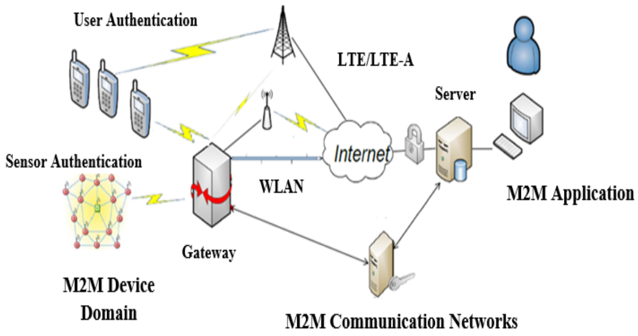


Figure. 1 Network architecture

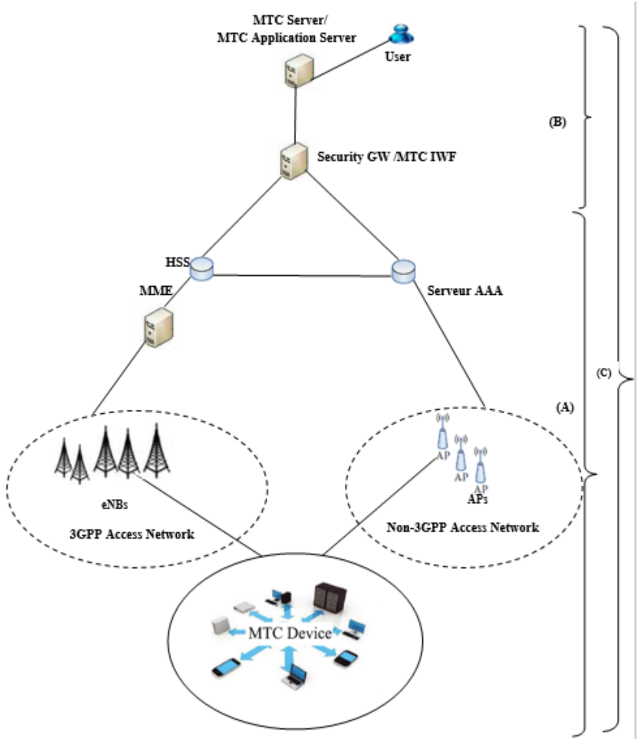


Figure. 2 Security architecture for M2M

(A2) Security between the M2M device and the core network through non-3GPP access network.

(B) Security for M2M communication between the 3GPP network and the MTC server/application can be further divided into security aspects when the MTC server is within the 3GPP network and when it is outside the 3GPP network.

(C) Security for M2M communication between the MTC server/application and M2M device.

2.3 Several new security issues and existing solutions

In this section, we focus on building more secure M2M communications in LTE networks by discussing several new security issues and presenting some solutions in all areas domain of the M2M architecture. The main contributions in this part are three-fields [9]:

Firstly, the authentication between the MTCD and the Evolved Packet Core (EPC) through 3GPP or non-3GPP access network (e.g., E-UTRAN, WLAN, or WiMAX) can be performed by Authentication and Key Agreement (AKA) protocols: Evolved Packet System (EPS-AKA) and Extensible Authentication Protocol (EAP-AKA) [10]. However the challenge of M2M research is authentication by the group when a large number of MTC devices simultaneously accessing the network will cause severe authentication signalling congestion. Once a huge number of MTC devices get connected to the network, each device must implement an independent access authentication process according to the standard protocols, which will cause serious traffic congestion in the Long Term Evolution (LTE) network. The current EPS-AKA and EAP-AKA are not suitable for group authentication; they need to be modified to apply to the group authentication of MTC. In a previous work [11, 12] we proposed two new group access authentication schemes, by which a huge number of MTC devices can be simultaneously authenticated by the network and establish an independent session key with the network respectively. Experimental results show that the proposed scheme can achieve robust security and avoid signalling overload on LTE networks.

Secondly, According to 3GPP TR 33.868 [13], the security Gateway (GW) can perform access control functionality to prevent the unauthorized MTC Server (MTCS) from accessing the EPC. It can authenticate with MTCS on behalf of the 3GPP network operator. The Network Domain Security (NDS/IP) security mechanism or private protection mechanism can protect the trigger indication sent from the MTCS to the security GW [14].

Thirdly, we assume the proxy signature technique together with several signature scheme to design a secure and efficient authentication and key agreement (AKA) protocol between the MTCD and MTCS. After a successful mutual authentication, a trust relationship can be built between the MTCD and MTCS; meanwhile, an end-to-and secure channel can be established between them.

According to this study, it can be observed that the proposed solutions to secure the different blocks of M2M architecture can guarantee a good level of security. However, the connection between the smart objects including a group of sensors, tablets, smartphones, etc. in M2M domain devices is insecure; therefore, there are distrustful relationships between M2M devices and gateway. The main subject of this work is to guarantee a level of security in M2M device domain without limited in mobile devices but also in sensor nodes.

To address this problem, we propose a new model include different authentication and security mechanisms in M2M device domain that can support recent technologies such as D2D communication and WSN.

3. Proposed mechanisms to secure group communication in M2M device domain

In this work, we focus to improve security links between M2M devices by proposed a new model include different schemes with multiple algorithms and protocols. These choices of mechanisms enable the sequence of events shown in Fig. 3 to be performed whenever the gateway (In this case we assume that the gateway is a device mobile) establish key exchange with the nodes and devices, in this procedure only 2 devices are shown for simplicity, it could be a greater number and the same thing for sensors.

First the gateway (Device A) will initiate the key generation with device B using an improved and efficient proposed D2D authentication based on the key agreement protocol of Bluetooth. The gateway can initiate the key exchange not only with other mobile devices but also with sensor which is considered the leader of group sensors using Diffie-Hellman as shown in Fig. 3. The communication between nodes sensors using TESLA can supports sensor-to-sensor and sensor-to-group authentication and communication. In the following, we will detail this procedure with a study of each mechanisms and their security analysis.

4. Secure group devices in D2D communication

D2D communication is a new paradigm in cellular networks. It allows users in close proximity to communicate using a direct link rather than having their radio signal travel all the way through

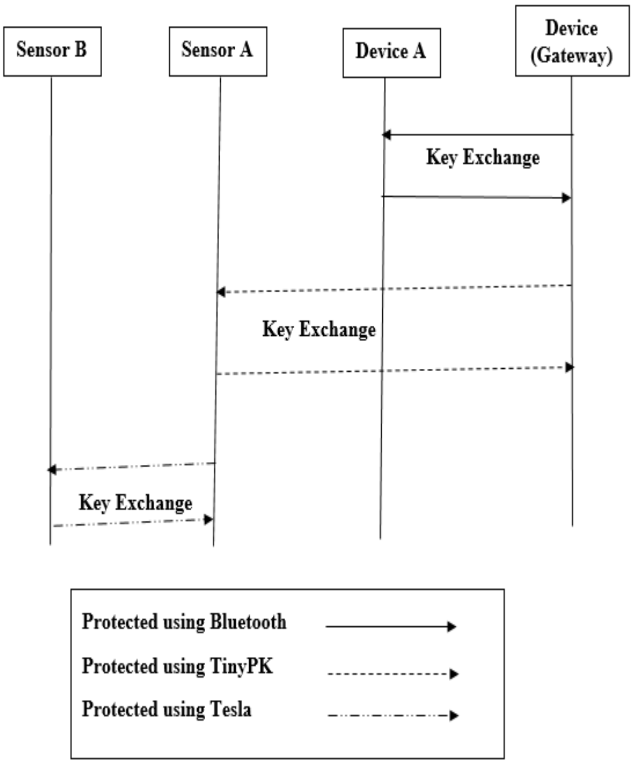


Figure. 3 Diagram of key exchange

the base station (BS) or the core network. One of its main benefits is the ultra-low latency in communication due to a shorter signal traversal path. Various short-range wireless technologies like Bluetooth, WiFi Direct and LTE Direct (defined by the 3GPP) can be used to enable D2D communication [15, 16].

However, we observed that most of the current D2D protocols such as Bluetooth and Wi-Fi Direct are not scalable and vulnerable to MITM, replay attacks and Denial of Service (DoS) attacks.

Bluetooth is used for transferring data and short-range communication using low power. Bluetooth devices authenticate each other through the pairing process. Pairing checks whether each device is authenticated and if authenticated, it allows the devices to generate a common link key [17].

The key agreement protocol is a crucial part of the security architecture of Bluetooth. Suppose that two Bluetooth devices, called A and B, want to communicate securely (in the rest of this paper, we will assume that A initiates the communication). Initially these devices do not share a secret. They perform a key agreement protocol to generate a link key and an encryption key [18].

In this section, we propose a D2D authentication protocol based in a mechanism for the key agreement protocol of Bluetooth. This improved protocol is efficient and secure by exploiting Attribute-Based Encryption (ABE) to secure the way to share initial secret keys among communicating parties and to solve the dependency of the keys on the PIN.

4.1 Attribute-based encryption

The concept of ABE is a relatively recent mechanism that combines asymmetric encryption with access control [19].

In an ABE system, the keys and encrypted messages of a user are tagged with a set of attributes and a particular key can decrypt an encryption only if there is a match between the attributes of the encrypted and the key of the user.

Attribute encryption is divided into two main classes. The first type associates CP-ABE that allows a user to decrypt the message if the attributes associated with his identity are those that respect the defined access policy for a message.

For the second type, the access policy is encoded in the secret keys of the users Key-Policy Attribute-Based Encryption (KP-ABE), the encryption is defined with respect to a set of attributes. Linking data confidentiality with a fairly expressive access control policy, attribute-based encryption can be an effective and interesting solution to overcome the disadvantages of traditional Public Key Infrastructure (PKI) based encryption mechanisms and allows to solve open security issues related to distributed environments.

4.2 Proposed scheme

We propose a D2D authentication protocol, which is secure in a mobile network environment. The proposed scheme is constructed based on Bluetooth authentication protocol by adding additional initial key sharing process by exploiting attribute-based encryption to secure the way to share initial secret keys among communicating parties. It enables scalable and secure initial key establishment even in a mobile network environment. In addition, the proposed protocol is also modified in such a way that it is secure against replay and modification attack by malicious relaying nodes.

A device may need to communicate either with an arbitrary mobile device, or with a specific group composed of multiple devices. In consideration of our scenario, we assume that the attribute keys are distributed to each device during the initial setup phase before the proposed authentication protocol.

D2D protocol proposed supports device-to-device and device-to-group authentication and communication. As mentioned above, Bluetooth has to share PIN before pairing, however, in a mobile network environment, it is difficult to guarantee the confidentiality and integrity of sharing secret information through D2D communication. Therefore, our D2D proposed exploits CP-ABE which enables a sender to define an access control policy and enforce it to the encrypted data. Thus, the sender can selectively distribute the PIN to a set of selected receivers in a scalable and secure way. Additional random number and Message Integrity Code (MIC) are adopted in the protocol to enhance integrity and confidentiality of authentication messages.

Fig. 4 shows our D2D authentication and key agreement procedure that knows five phases to execute the procedure named as:
Phase 1- Establishment of the initialization key, Phase 2- Signature Generation and Verification, Phase 3- Link/Session key generation, Phase 4- Mutual entity authentication and Phase 5- Data exchange.
The protocol progresses as follows:

Phase 1- Establishment of the initialization key :
1. User A enters PIN to device A
2. A → B: CP-ABE (TB, PIN)

Device A defines access policy TB with a set of attributes, encrypts PIN under TB, and sends it to device B. Device A generates 128 bits random number. Then it generates initial key Kinit using hash function H() on inputs RN and PIN.

3. A → B: RN, MIC (PIN $\oplus$ GI_B $\oplus$ GI_A, RN)

Device A sends RN in plaintext and MIC of RN generated with a key which is XOR of A's device information GIA, B's device information GIB, and PIN to device B. Device B decrypts PIN from the cipher text if a set of attributes of B satisfies the access policy TB. Then, user B enters the PIN. Finally, B generates MIC of received RN with a key that is PIN $\oplus$ GI_B $\oplus$ GI_A. If the MIC from B is equal to MIC from A, B can generate accurate initial key Kinit using hash function H on inputs RN and PIN.

Device-to-group authentication is almost the same as the above protocol except that GI_B is replaced by another group information such as a group ID.

Phase 2- Signature generation and verification :

The tow devices have the same PIN but they have not yet verified. Each one choose another random number RN_A for device A and RN_B for B and send this numbers to each other. Based on the secret key, random number and public keys, pairing devices computes their digital signature, and substantiates digital signatures of each other. The protocol proceeds further, only when signatures are verified. Here g () is Keyed-Hash Message Authentication Code Secure Hash Algorithm (HMAC- SHA256) function.

Phase 3- Link/Session key generation :

The unit key of device A is the link key, it is transmitted encrypted from A to B. This encryption is done by XOR'ing the unit key of A with the initialization key.

If the link key is a combination key, then both devices first generate a random number LK_RAND. These random numbers are encrypted with the initialization key and sent to the other device. Now they both calculate LK_KA = F(LK_RANDA;ADDRA) and LK_KB = F(LK _RANDB;ADDRB). The combination key KAB is the XOR of LK_KA and LK_KB. F (…) is HMAC-SHA256 function. After the generation of the link key, the (old) initialization key is definitively discarded and a mutual authentication is started with the exchanged link key that is shared between both devices.

Phase 4- Mutual entity authentication :

Each time a new shared key and a link key are generated, both devices perform a mutual authentication protocol. The authentication scheme is based on a challenge-response protocol. This protocol is performed twice. First, B authenticates itself to A, if this authentication is successful, the roles are switched (B becomes the verifier and A the prover). The description of this procedure is the same as original Bluetooth protocol.

Phase 5- Data exchange :

Data exchange is finally the easiest part of Bluetooth communication. Once they are authenticated, the two devices exchange a communication key, generated by algorithms from the Kauth authentication key and a random value.

4.3 Security analysis

In this paragraph, we analyze the security of our proposed protocol against some attacks: man-in-the-middle attack, DoS attack and replay attack.

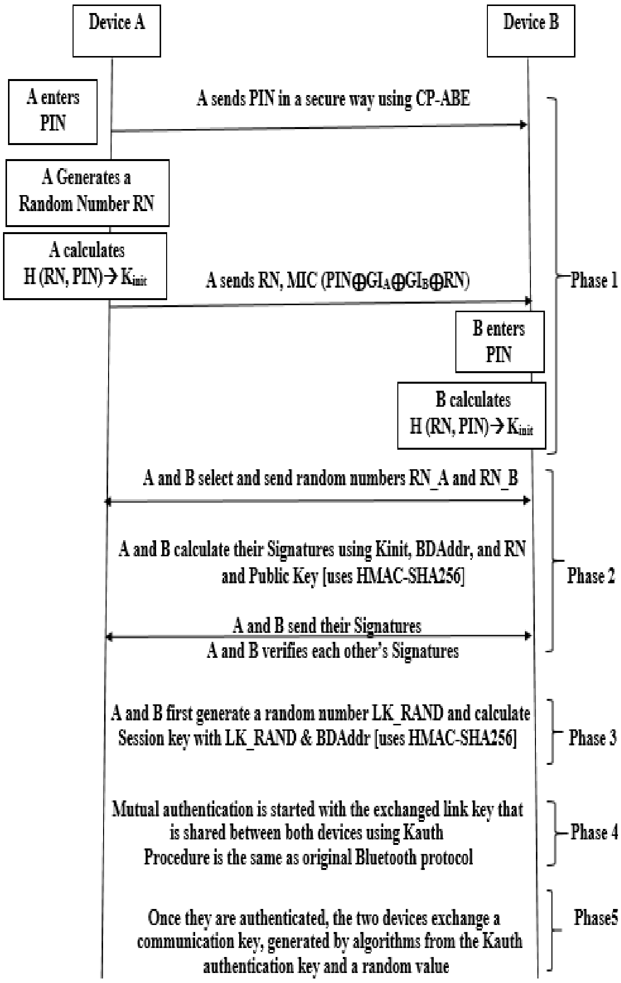

Device A
Device B
A enters PIN
A sends PIN in a secure way using CP-ABE
A Generates a Random Number RN
A calculates H (RN, PIN)→ Kinit
A sends RN, MIC (PIN⊕GIA⊕GIB⊕RN)
B enters PIN
B calculates H (RN, PIN)→ Kinit
Phase 1
A and B select and send random numbers RN_A and RN_B
A and B calculate their Signatures using Kinit, BDAddr, and RN and Public Key [uses HMAC-SHA256]
A and B send their Signatures
A and B verifies each other's Signatures
Phase 2
A and B first generate a random number LK_RAND and calculate Session key with LK_RAND & BDAddr [uses HMAC-SHA256]
Phase 3
Mutual authentication is started with the exchanged link key that is shared between both devices using Kauth
Procedure is the same as original Bluetooth protocol
Phase 4
Once they are authenticated, the two devices exchange a communication key, generated by algorithms from the Kauth authentication key and a random value
Phase5

Figure. 4 D2D proposed authentication procedure

4.3.1. Man in the middle attack

In our proposed D2D scheme, when a sender device transmits the secret information, such as PIN, the device sends it after encrypting it using CP-ABE algorithm. It ensures that even if malicious nodes relay the authentication exchanges in route, they cannot obtain any secret information as long as their attributes do not satisfy the access policy embedded in the ciphertext. Thus, end-to-end confidentiality is guarantee against individual or colluding attack of malicious nodes. Therefore, sharing PIN is secure under man-in-the-middle attack. Also, message integrity is preserved due to the adoption of MIC for a random number, which is generated with a securely shared PIN.

4.3.2. DoS attack

The use of a blacklist was included in the Bluetooth security architecture to avoid repetitive authentication attempts in which an attacker can verify a guess of the PIN. After a relatively short amount of time, the attacker would find the correct PIN. This was avoided by the use of a blacklist. The problem is that this list can be abused in a DoS attack. Other Denial of Service attacks are still possible (e.g., the sleep deprivation attack), but this cannot be prevented in mobile networks.

4.3.3. Replay attack

In the mobile network, replay attack is done by the relay node. In order to prevent the replay attack, we adopt random number. When the PIN is shared, a receiver obtains random number from MIC, it can know whether the message is replayed or not by comparing its own number and sender's number. Thus, it is secure under the replay attack.

4.4 Verification

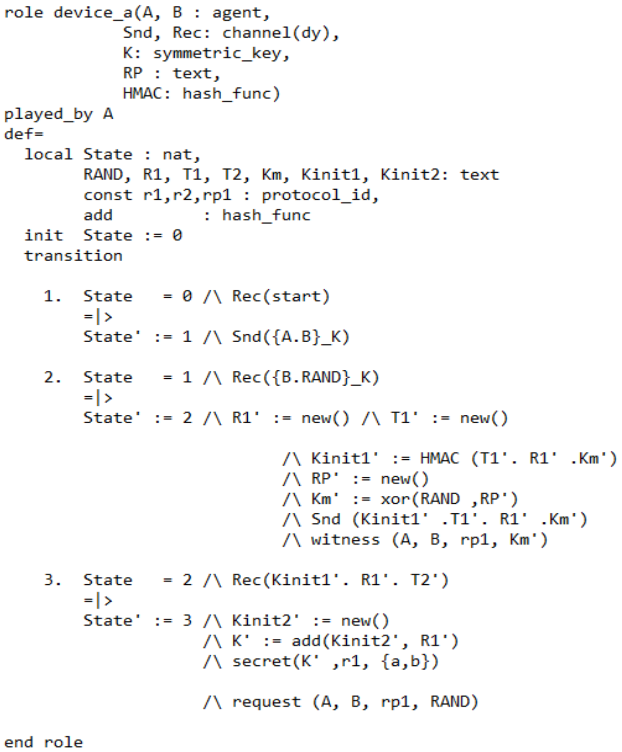
This solution was checked by the security protocol verification tool AVISPA that indicated that it is a very secure level. The main advantage of this tool is the ability to use different verification techniques on the same protocol specification. The protocol designer interacts with the tool by specifying a security problem in the High Level Protocol Specification Language (HLPSL). The HLPSL is an expressive, modular, role-based, formal language that is used to specify control-flow patterns, data-structures, alternative intruder models and complex security properties, as well as different cryptographic primitives and their algebraic properties [20].

The primary goal of our proposed protocol is to provide mutual authentication services between two devices.

In our proposed scheme described with HLPSL Language, the MTC devices A and B represent the two participants in basic roles.

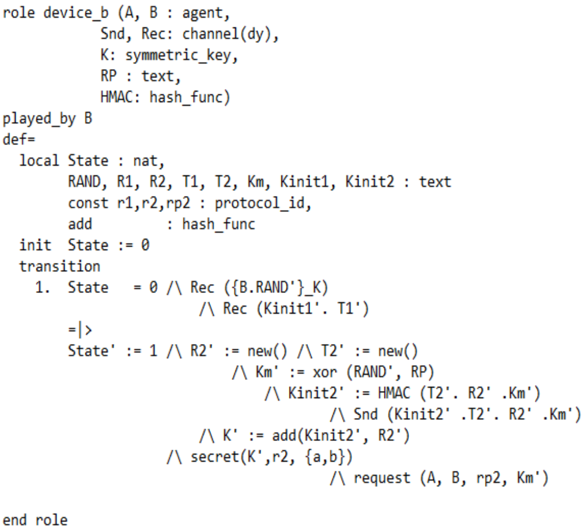
We need to verify that the proposed protocol can provide a successful mutual authentication between these two MTC devices by using back-end servers.

In this paper, we only present the authentication analysis of two MTC device, basic roles of the MTC device A and MTC device B and the authentication goals are shown in Figs. 5, 6 and 7, respectively.



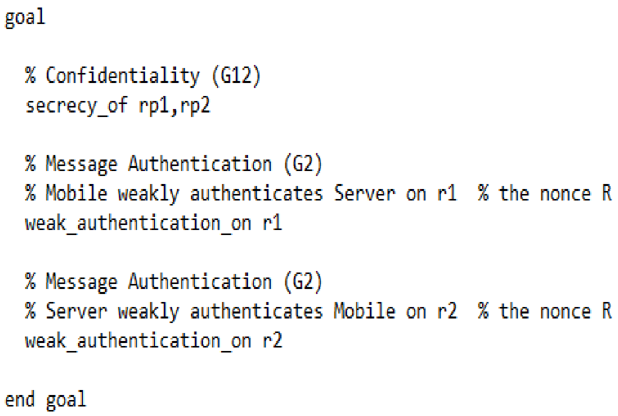
```
role device_a(A, B : agent,
            Snd, Rec: channel(dy),
            K: symmetric_key,
            RP : text,
            HMAC: hash_func)
played_by A
def=
  local State : nat,
        RAND, R1, T1, T2, Km, Kinit1, Kinit2: text
        const r1,r2,rp1 : protocol_id,
        add          : hash_func
  init  State := 0
  transition

    1.  State   = 0 /\ Rec(start)
        =|>
        State' := 1 /\ Snd({A.B}_K)

    2.  State   = 1 /\ Rec({B.RAND}_K)
        =|>
        State' := 2 /\ R1' := new() /\ T1' := new()

                            /\ Kinit1' := HMAC (T1'. R1' .Km')
                            /\ RP' := new()
                            /\ Km' := xor(RAND ,RP')
                            /\ Snd (Kinit1' .T1'. R1' .Km')
                            /\ witness (A, B, rp1, Km')

    3.  State   = 2 /\ Rec(Kinit1'. R1'. T2')
        =|>
        State' := 3 /\ Kinit2' := new()
                    /\ K' := add(Kinit2', R1')
                    /\ secret(K' ,r1, {a,b})

                    /\ request (A, B, rp1, RAND)

end role
```

Figure. 5 Role of device A



```
role device_b (A, B : agent,
           Snd, Rec: channel(dy),
           K: symmetric_key,
           RP : text,
           HMAC: hash_func)
played_by B
def=
  local State : nat,
        RAND, R1, R2, T1, T2, Km, Kinit1, Kinit2 : text
        const r1,r2,rp2 : protocol_id,
        add         : hash_func
  init  State := 0
  transition
    1.  State   = 0 /\ Rec ({B.RAND'}_K)
                     /\ Rec (Kinit1'. T1')
        =|>
        State' := 1 /\ R2' := new() /\ T2' := new()
                        /\ Km' := xor (RAND', RP)
                           /\ Kinit2' := HMAC (T2'. R2' .Km')
                                 /\ Snd (Kinit2' .T2'. R2' .Km')
                     /\ K' := add(Kinit2', R2')
                  /\ secret(K',r2, {a,b})
                                 /\ request (A, B, rp2, Km')

end role
```

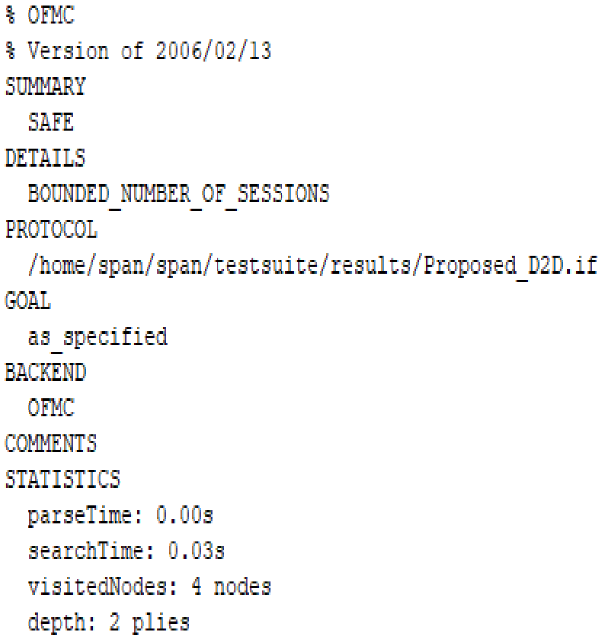
Figure. 6 Role of device B



```
goal

  % Confidentiality (G12)
  secrecy_of rp1,rp2

  % Message Authentication (G2)
  % Mobile weakly authenticates Server on r1  % the nonce R
  weak_authentication_on r1

  % Message Authentication (G2)
  % Server weakly authenticates Mobile on r2  % the nonce R
  weak_authentication_on r2

end goal
```

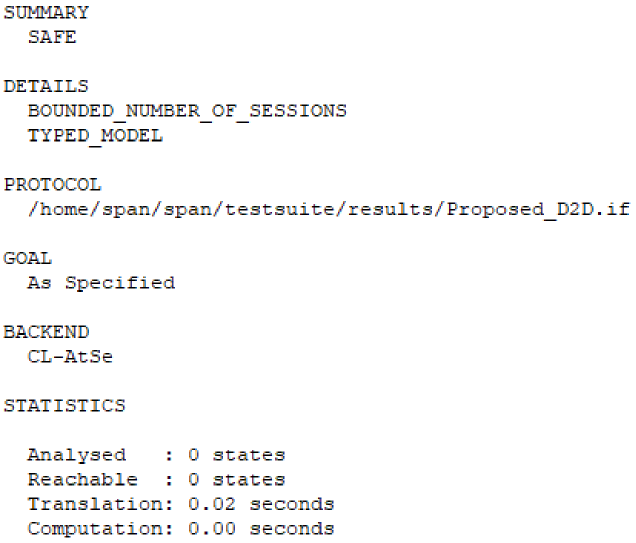
Figure. 7 Analysis goals of our scheme

The back-end On-the-Fly-Model-Checker (OFMC) and CL–based Attack Searcher (CL-AtSe) will be used to verify that the proposed scheme maintains its security objectives even under various attacks. We run the Security Protocol Animator (SPAN) for AVISPA in OFMC and CL-AtSe modes to validate the above goals. The output of the model checking results is shown in Figs. 8 and 9. According to this figures, we can conclude that our scheme can achieve the security goals and withstand various attacks including MITM attacks, impersonation attacks, DoS and replay attacks under the test of AVISPA and SPAN using the OFMC and CL-AtSe back-ends with a bounded number of sessions



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/Proposed_D2D.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.03s
  visitedNodes: 4 nodes
  depth: 2 plies
```

Figure. 8 Results reported by the OFMC back-end in SPAN



```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/Proposed_D2D.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed    : 0 states
  Reachable   : 0 states
  Translation: 0.02 seconds
  Computation: 0.00 seconds
```

Figure. 9 Results reported by the CL-AtSe back-end in SPAN

Table 1. Communication cost

Protocols	**Communication Cost**
Bluetooth Standard	RN
SSP Protocol	RN + K_ECDH
Hybrid Algorithm	RN + CT
Proposed	CT + RN + SMIC

RN: Random Number, SMIC: MIC Size, CT: CipherText, K_ECDH: Elliptic Curve Diffie–Hellman Key.

4.5 Performance Analysis

In this section, we compare existing protocols based Bluetooth such as the Standard Bluetooth protocol, Secure Simple Pairing SSP protocol [21] and a Hybrid Algorithm [22] with our proposed scheme in terms of computation and communication cost [23].

4.5.1. Communication cost

In the case of existing protocols such as Bluetooth protocol, SSP protocol and a Hybrid Algorithm, the communication cost is demanded only a random number size and Cipher text, on the other hand our proposed protocol requires additional communication cost for PIN sharing securely and for sending MIC. This additional cost in our scheme is introduce to make the protocol more secure against any attacks and adversaries in the network.

Table 1 shows the analysis results in terms of communication.

4.5.2. Computation cost

In the proposed scheme, compared to existing protocols, CP-ABE encryption and decryption are additionally introduced in order to send PIN securely in a network. In addition, computation for MIC generation is required to guarantee message integrity. Even if it adds additional computation cost, message integrity is guaranteed, which cannot be preserved in Bluetooth.

We took advantage of the Crypto++ Library [24] to measure the elapsed time of the cryptographic operations. The measurement ran on an Intel Core Duo 1.86 GHz and 2 gigabyte RAM under an Ubuntu 11.10 operating system. Table 2 demonstrates the average elapsed time of some cryptographic operations. During the pairing phase, Bluetooth standard and Hybrid Algorithm will cost respectively: 3P+6H+2SE+2SD = 1181 us, SE+ASE+SD+ASD=1862us. While the proposed scheme will cost :
CP_ABE+CP_ABD+3P+7H+3SE+2SD = 9409 us. Fig. 10 shows the comparison result of the computation cost.

Table 2. Average elapsed time of cryptographic operations used in comparing computational delays

Operation	Symbol	Time (µs)
HMAC-SHA-256	H	67
Pseudo random generator	P	45
Symmetric encryption/decryption(AES-256)	SE/SD	161
Asymmetric Encryption (RSA)	ASE	80
Asymmetric Decryption (RSA)	ASD	1460
Multiplication over elliptic curve	T_{mul}	612
Addition over elliptic curve	T_{add}	125
Ciphertext-Policy Attribute-Based Encryption/Decryption	CP-ABE/ABD	4000
Signature (DSA)	S_DSA	450
Verification (DSA)	V_DSA	520

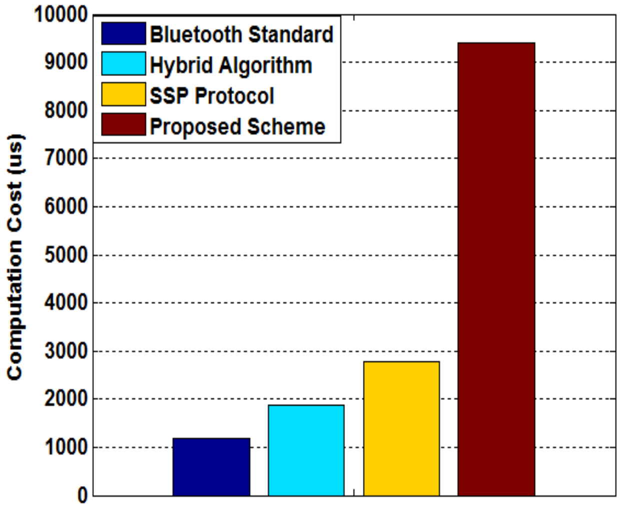


Figure. 10 Computation cost

5. Secure Group Communication in Wireless Sensor Network

A wireless network is a type of network where various physical devices (e.g. computer, laptops, PDAs etc) are interconnected with each other using network infrastructure. Owing to wireless medium of data communication, the security risk is potentially high for unauthorized access and intrusion of various malicious programs. The security protocols of wireless network are governed by family of IEEE 802.11 standards. Wireless Network is studied in research with respect to wireless LAN (Local Area Network), wireless mesh network, wireless sensor network, mobile adhoc network, etc. In recent times, wireless sensor network was on constant focus among the research community owing to its potential advantage of data collection in remote areas as well as security problems associated with it. WSN consists of various sensor motes that form a cluster and perform data aggregation. Usually, the aggregated data is forwarded from the sensor nodes to the base station, which then reaches to user for analysis. The security problems is a matter of concern even for wireless sensor network that aims for either compromising the routing protocols or invoke illegitimate access to resources by bypassing the security protocols. In a wireless sensor network, the communication takes place by group based, where sensor nodes are deployed in groups and each group performs communication using keys [25, 26].

This part examines the area of authentication for sensor networks by make a combination between algorithms, which could be used. This combination is recommended because TinyPK is able to enable the communicating nodes by creating a secret shared key. In order to secure, the link between sensors it is recommended that TESLA is used as it allows the key exchange is verified and can achieve a high levels of security.

5.1 Secure the link device to sensor using TinyPK

TinyPK is focused on supporting confidentiality and authentication for wireless sensor networks. The protocol uses a Certification Authority (CA) whose public key has to be preloaded onto the nodes in the networks, meaning that the nodes will require some pre-configuration before they can be deployed in the field this can pose scalability problems. However, TinyPK does not use certificates, as there is no real-time access to the CA, this poses problems when it comes to revoking keys.

The protocol operation is divided into two parts: the external party authentication and the node authentication. In the first part, the external party authenticates itself by means of a challenge. After this phase, the node and the external party share the network key and a nonce. In the second phase, the node and the external party establish a new key pair by means of Diffie-Hellman exchange. Each node has a static Diffie-Hellman key, while the third party generates an ephemeral Diffie-Hellman key. A node credential is also exchanged in order to authenticate the node. The messages sent between the nodes to establish the keys are shown in Fig. 11; once this is complete the key can be calculated according to this formula $Key = (g^{R1} \bmod P)^{R2} \bmod P = (g^{R2} \bmod P)^{R1} = g^{R1*R2} \bmod P$.

5.2 Securing sensors using TESLA

In multicast security architectures, the group control dictated by previously defined security policies requires authentication of the members or authentication of the source or both at the same time. The authentication of the members is carried out via methods using access control lists and certificates capable of mutually and individually authenticating the issuer and the receiver. This brings us back to the point-to-point authentication, aimed at ensuring a node the real identity of its interlocutor [27].

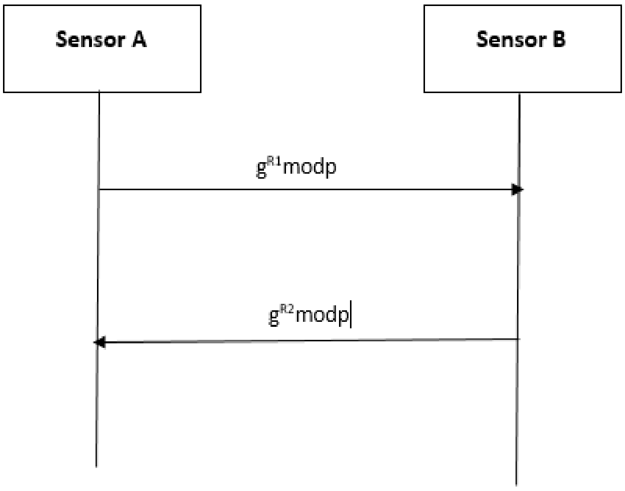


Figure. 11 Key exchange for Diffie-Hellman as used in TinyPK

TESLA is a commonly used protocol for broadcast authentication in wireless networks. TESLA is based on symmetric Message Authentication Codes (MAC), but introduces the element of asymmetry by delaying the disclosure of secret key.

The data flow in TESLA is unidirectional: the data flows only from the source to the receivers. This implies that the extra cost of the authentication of the source is independent of the number of receivers.

The basic concepts of TESLA are based on [28]:

1. The time slots: each packet Pi is authenticated separately, with MAC. Time is divided into t intervals of time Tint each. The transmitter can send 0 or more packets per slot Ij. At each interval Ij corresponds to it an authentication key k'j.
2. MAC keys: the transmitter generates a string of keys, k1, k2, ..., kt using a one-way function. The last key of the chain kt is first generated randomly and the others are derived using the equation: kj-1 = f(kj). Then, the transmitter generates the MAC keys k'j = g (kj) with g another one-way function. From this architecture flows an important property of TESLA which is the tolerance to packet loss. Indeed, even if all the packets sent in a given interval are lost (and consequently all the keys revealed at this interval), the receivers will be able to authenticate the packets based on future intervals. For example, packets sent during the interval Ij can be authenticated even if all the packets during the interval Ij+d (where kj is revealed) are lost. d is the timeframe for the key statement presented below.
A receiver can always calculate kj from any key kj+m with m>=d.
3. Validation of the keys of the chain: the source can validate the keys of the chain by signing a packet containing a key of the chain, or by including this key in an authenticated packet. For example, to validate chain keys k1, k2, ..., kt, the source can send an authenticated packet containing the key k0 = f(k1).
4. Time of synchronization between transmitters and receivers: the receivers need to know a limit superior of the time of the source. Thus, if the time difference between the source and the transmitter is dt, we suppose that the receivers know a Dt such that Dt >= dt
5. Time of receipt of keys or delay: this indicates the time (number of intervals) that the receiver needs to be able to authenticate a packet in an interval Ij. This delay has a direct consequence on the space required storage capacity.

Table 3.Context and features of methods

Approaches	Context	Functionality
Key Agreement	Restricted set of nodes	Share a secret (session key) and establish secure multicast communications (1 to n or n to n)
TESLA	An unreliable multicast data stream from a source to receivers (1 to n)	Source authentication Allow scalability Low additional cost of calculation and communication Robustness and tolerance to packet loss
Authentication using FEC	An unreliable multicast data stream from a source to receivers (1 to n)	Authentication and not repudiation of the source Tolerance to data loss Ensure data integrity Real time broadcasting Low additional cost of communication

Revaluation time d is crucial in TESLA. Indeed, the choice of a small d makes it difficult to authenticate packets by receivers far enough from the source. On the other hand, the choice of a large d would require a storage space rather large number of receivers. For this, there are other authentication methods in TESLA:

- Immediate authentication: TESLA proposes a mechanism allowing an immediate authentication of packets by the receivers. Each source-rejected package contains a hash of the future packet. This the method involves an overhead at the packet size level. In addition, it is no longer robust against the losses of packets.
- Heterogeneous receivers: Receivers close to the source prefer a short key response time to be able to take advantage of a short authentication period. On the other hand, receivers far enough away will not be able to operate with a short notice period because the packet transmission time may exceed this delay and therefore the security requirements will be violated and the packets will not be able to authenticate.

5.3 Security and efficiency analysis

In Table 3, we present a summary of TESLA and different authentication methods of the source [29, 30], presented above. The table defines the context and objectives of each method.

To be able to judge the appropriateness of these methods within the WSN networks, we have defined the following criteria for analysis:

Robustness: the ability of the authentication architecture to respond to data loss. The Key Agreement approach does not provide solutions against data loss. These losses can to be problematic especially during the initial phase of determining the session key.

On the other hand, authentication with TESLA is robust against packet loss (except for the case of immediate authentication with TESLA, where each issued packet contains the hash of the next packet).

Like TESLA, authentication with FEC is robust against packet loss. FEC also allows correction of data loss.

Accessibility: The ability of receivers to access the multicast flow reception service and to authenticate packets from any point in the stream. Accessibility is provided by authentication with FEC, more difficult by TESLA, because of its phase of synchronization and initialization.

For "Key Agreement" protocols, accessibility is very low because the session key must be recalculated to take into account the new member's contribution.

Data storage: the maximum number of packets that the source or receivers must store.

"Key Agreement" does not require memory storage, neither source nor quoted receivers.

Authentication according to TESLA does not require memory storage source side. Receivers, on the other hand, must store packets for periods of time in the worst case. (d being the key reveal time in TESLA).The authentication using FEC, according to its alternatives, requires or not a storage in memory of the data. For ECU (Unbuffered Sender Scheme), no memory storage is needed from the issuer's side. On the other hand, the receivers must

Table 4. Storage of data

Approaches		Data storage	
		Source	Receivers
Key Agreement		0	0
TESLA		0	Packages received in intervals
FEC	0	2b	ECU
	2b	0	ECU2
	0	0	ECU1

Table 5. Authentication delay

Approaches		Authentication delay
Key Agreement		4
TESLA		Packets received at intervals +1
FEC	ECU	2b
	ECU2	0
	ECU1	0

store at most two data blocks. For EC2 (Double Buffered Scheme), on the side of the receivers, authentication is done automatically without storage in memory. While the source has to store two blocks of data at a time. For EC1 (Single Buffered Scheme), no storage is needed (b being the number of packets per block). The table 4 summarizes these comparisons.

Authentication delay: the maximum number of packets that the receivers must receive in order to authenticate the first packet.

According to the "Key agreement" protocol, a node must wait for the reception of 4 packets so that the session key is established.

For authentication using TESLA, at most, receivers must wait for the receipt of the first initialization packet, plus the number of packets during time slots.

Authentication using FEC requires, for its ECU alternative, a maximum latency equal to 2 * number of packets in a block. (Table 5)

Cost in terms of computing power :

"Key Agreement" requires an encryption / decryption operation of each packet sent by the source, as the confidentiality of the data is also ensured. Encryption and decryption operations are done with the same session key.

Authentication with TESLA requires the signature (source side) and the verification (receiver side) of the first protocol initialization packet. Then, the source calculates a packet hash function.

Authentication using FEC requires source dimension and for each block, b hash operations (b being the number of packets per block), a digital signature and two encryption-decryption operations. From the side receivers, and at a minimum, hash operations and a signature check are also required.

Overhead in terms of bandwidth:

"Key Agreement" has no impact on bandwidth. Indeed, the authentication is done outside the sending of secure multicast data (out of band).

For TESLA, the bandwidth overhead may not exceed 10 bytes per packet. However, according to the hash functions used, this number may vary. For example, using 80-bit HMAC-MD5, we reached 24 bytes per packet (10 bytes for the key checked, 10 bytes for the authentication information MAC and 4 bytes for the interval index).

The overhead for the solution using FEC is dependent on b (number of packets per block) and p (rate of losses per block).

Synchronization between source and receivers: Only authentication using TESLA requires synchronization between source and receivers.

6. Conclusion

This paper discussed security procedures for M2M devices, started with the description of a general security architecture along with its basic procedures and presented several sew security issues and existing solutions. Our goal in this work is to propose two approaches for secure group communication over D2D links and wireless sensor network.D2D communication is getting lots of attention due to its applicability in mobile network environment. However, current D2D authentication protocols cannot be used because they are vulnerable to inside attacks such as MITM attack or replay attack by relaying nodes. In this paper, we proposed a D2D authentication protocol using CP-ABE to solve the problems with regard to sharing the initial secret information safely under the attacks.

Even if the proposed protocol are designed on the basis of Bluetooth protocol, our scheme solve the initial key establishment problems, integrity problems in the presence of the inside adversaries in networks and offers best performance than the standard Bluetooth. Therefore, the proposed scheme can be applicable to the other D2D protocols, such as Wi-Fi Direct. In addition, this paper has examined the area of authentication for sensor networks by make a combination between algorithms which could be used. This combination is recommended because TinyPK is able to enable the communicating nodes by creating a secret shared key. In order to secure the link between sensors it is recommended that TESLA is used as it allows the key exchange is verified and can achieve a high levels of security. For future work, we will be interested in an in-depth performance analysis as well as simulation tests to show the practicability and efficiency of our model.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey", *Computer Networks*, pp.2787–2805, 2010.

[2] D. C. Yacchirema and C.Palau, "Smart IoT Gateway For Heterogeneous Devices Interoperability", *IEEE Latin America Transactions*, Vol. 14, No. 8, 2016.

[3] A. Riker, T. Cruz, B. Marquesy, M. Curado, P. Simoes, and E. Monteiro, " Efficient and Secure M2M Communications for Smart Metering", In: *Proc. of the 2014 IEEE Emerging Technology and Factory Automation*, 2014.

[4] J. Wan, D. Li, C. Zou, and K. Zhou, "M2M Communications for Smart City: An Event-Based Architecture", In: *Proc. of the 2012 IEEE 12th International Conference on Computer and Information Technology*, 2012.

[5] J. Kim, J. Lee, J. Kim, and J. Yun, "M2M service platforms: Survey, issues, and enabling technologies", *IEEE Communications Surveys Tutorials*, Vol.16, No.1, pp. 61-76, 2014.

[6] T. Taleb, and A. Kunz, "Machine type communications in 3GPP networks: Potential, challenges and solutions", *IEEE Communications Magazine*, Vol.50, No.3, pp.178-184, 2012.

[7] D. Astely, E. Dahlman, G. Fodor, S. Parkvall, and J. Sachs, "LTE release 12 and beyond," *IEEE Communications Magazine*, Vol. 51, No. 7, pp. 154–160. 2013.

[8] 3GPP TS 22.368. V14.0.1, "Service requirements for Machine-Type Communications (MTC); Stage 1 (Release 14)", 2017.

[9] C. Lai, H. Li, Y. Zhang, and J. Cao, "Security Issues on Machine to Machine Communications", *KSII Transactions on Internet and Information Systems*, Vol. 6, No. 2, 2012.

[10] A. H. Hassanein, A. A. Abdel Hafez, and A. A. Gaafar, "New Authentication and Key Agreement Protocol for LTE-WLAN Interworking," *International Journal of Computer Applications*, Vol. 61, No. 19, 2013.

[11] M. Ouaissa and A. Rhattoy, "A New Scheme of Group-based AKA for Machine Type Communication over LTE Networks", *International Journal of Electrical and Computer Engineering*, Vol. 8, No. 2, pp. 1169-1181, 2018.

[12] M. Ouaissa, A. Rhattoy, and M. Lahmer, "Group Access Authentication of Machine to Machine communications in LTE Networks", In: *ICC'17 Proceedings of the Second International Conference on Internet of things and Cloud Computing*, 2017.

[13] 3GPP TR 33.868 V0.11.0, "Security aspects of machine-type communications", 2012.

[14] M. Sher and T. Magedanz, "Developing Network Domain Security (NDS) Model for IP Multimedia Subsystem (IMS)", *Journal of Networks*, Vol. 1, No. 6, 2006.

[15] M. Hausy, M. Waqas, A. Y. Dingy, and Y. Li, "Security and Privacy in Device-to-Device (D2D) Communication: A Review", *IEEE Communications Surveys and Tutorials*, Vol.19, No.2, pp. 1054-1079, 2017.

[16] M. Alam, D. Yang, J. Rodriguez, and R. A. Abd-Alhameed Secure Device-to-Device Communication in LTE-A, *IEEE Communications Magazine*, Vol. 52, No. 4, 2014.

[17] T. Kumar, "Improving Pairing Mechanism in Bluetooth Security", *Int. J. of Recent Trends in Engineering and Technology*, Vol. 2, No. 2, 2009.

[18] S. Gajbhiye, S. Karmakar, M. Sharma, and S. Sharma, "Design and analysis of pairing protocol for bluetooth enabled devices using R-LWE Lattice-based cryptography", *Journal of Information Security and Applications*, Vol. 35, No. C, pp.44–50, 2017.

[19] S. Moffat, M. Hammoudeh, and R.Hegarty, "A Survey on Ciphertext-Policy Attribute-based Encryption (CP-ABE) Approaches to Data Security on Mobile Devices and its Application to IoT", In: *Proc. of the International*

Conference on Future Networks and Distributed Systems, 2017.

[20] T. A. Team, "AVISPA v1. 1 User Manual 2006," http://avispaproject.org/.

[21] K. Haataja, K. Hyppönen, S. Pasanen, and P. Toivanen, *Overview of Bluetooth Security In: Bluetooth Security Attacks*, SpringerBriefs in Computer Science, Springer, Berlin, Heidelberg. 2013.

[22] K. Rege, N. Goenka, P. Bhutada, and S. Mane, "Bluetooth Communication using Hybrid Encryption Algorithm based on AES and RSA", *International Journal of Computer Applications*, Vol. 71, No.22, 2013.

[23] R.-H. Hsu and J. Lee, "Group Anonymous D2D Communication with End-to-End Security in LTE-A", In: *Proc. of the 2015 IEEE Conference on Communications and Network Security*, 2015.

[24] W. Dai, Crypto++ 5.6.0 Benchmarks, Available: http://www.cryptopp.com/benchmarks.html, 2009.

[25] H S Annapurna and M. Siddappa, "A Survey on Security Techniques in Group Communication for Wireless Sensor Networks", *International Journal of Computer Applications,* Vol.113, No. 7, 2015.

[26] K. Chaudhary and G. Shinde, "Survey on Group Authentication in Wireless Sensor Networks", In: *Proc. of the International Conference on Pervasive Computing*, 2015.

[27] B. Mbarek, A. Meddeb, W. B. Jaballah, and M. Mosbah, "An Efficient Broadcast Authentication Scheme in Wireless Sensor Networks", In: *Proc. of the 8th International Conference on Ambient Systems, Networks and Technologies*, 2017.

[28] K. Grover and A. Lim, "A survey of broadcast authentication schemes for wireless Networks", *Ad Hoc Networks*, Vol. 24, No. PA, pp. 288-316, 2015.

[29] A. Pannetrat and R. Molva, "Efficient multicast packet authentication", In: *Proc. of the 10th Annual Network and Distributed System Security Symposium*, 2003.

[30] N. Asokan and P. Ginzboorg, "Key-agreement in ad-hoc networks", *Computer Communications*, Vol.23, No.17, pp.1627–1637, 2000.