A Secured and Authorized SEEN Protocol for Mobile Multimedia Data Collection Scheme in WMSNs

V. Prakasham
Department of Computer Science and Engineering
KSR Institute for Engineering and Technology, Namakkal, Tamil Nadu, India

V. Sowmitha
Department of Computer Science and Engineering
KSR Institute for Engineering and Technology, Namakkal, Tamil Nadu, India

M.Vimaladevi
Department of Computer Science and Engineering
KSR Institute for Engineering and Technology, Namakkal, Tamil Nadu, India

***Abstract*—Wireless Multimedia Sensor Networks (WMSNs) produce enormous amounts of big multimedia data. Due to big data, Multimedia Sensor Nodes (MSNs) cannot store generated multimedia data for a long time. In this situation, mobile sinks can be used for information assortment. However, because of defenseless nature of wireless networks, there is a requirement for a productive security plan to validate both MSNs and mobile sinks. In this paper, we propose a scheme to protect an underlying WMSN during mobile multimedia big data collection. The proposed scheme is a two-layer scheme. At the primary layer, all MSNs are circulated into little clusters, where each cluster is spoken to by a solitary cluster Head (CH). At the second layer, all CHs verify identities of mobile sinks before sharing multimedia data and uses Secured and Authorized Selective Encryption (SEEN) protocol for assuring high data trustworthiness requires that the system satisfies two key security properties: confidentiality and integrity. We propose a Secured and Authorized SEEN method to secure big sensing data streams that satisfies the desired multiple levels of confidentiality and data integrity. We assess the exhibition of the proposed scheme through broad simulation results. The reenactment results demonstrate that the proposed scheme performs better when contrasted with existing state-of-the-art approaches as far as versatility and handshake span. The proposed scheme is likewise dissected as far as validation rate, data newness, and packet conveyance ratio, and has indicated a superior performance.**

***Index Terms*—Big data stream, selective encryption, data confidentiality, data integrity, WMSNs, multimedia, clusters**

I. INTRODUCTION

TRADITIONAL concept of the Internet of Things (IoT) consists of small battery-operated sensing devices that are unable to perform long-range wireless communication [1]. These sensing devices usually form a small Wireless Sensor Network (WSN), and forward sensed data to a nearby Base Station (BS). In WSNs, computationally-complex tasks are always performed at BS. In recent years, evolutionary steps have been taken in the field of WSNs where audio and visual sensors are integrated into simple sensor nodes and transformed them into Multimedia Sensor Nodes (MSNs). These nodes are able to store, correlate and fuse both multimedia and non-multimedia data [2].

The MSNs are gaining the attention of researchers across the globe due to their wide range of applications in daily lives, such as e-health, transport management system, and surveillance systems. For example, in smart cities, smart cameras installed at major junctions, roads and highways can capture current situation of vehicular traffic, and forward captured multi- media data to either local or remote cloud servers for further processing and analysis.

Data produced from a large variety of sources using sensing devices are streamed towards Data Stream Managers (DSM) for processing and decision making. This trend gives birth to an area, called big data stream [19], [20]. The verity of applications and data sources makes the need for data dependability such

that only trustworthy and dependable information is considered for decision making processes. Data security (i.e., more specifically ensuring data integrity and confidentiality) is an efficient and effective procedure to assure data trustworthiness/ dependability, since DSM processes the data streams in near real time and performs the data analytics; the appropriate actions are performed based on the results from the analytics. It is thus important that data trustworthiness is assured during the lifecycle of big data stream processing.

There are different security requirements for different emerging critical applications. Let's consider some applications such as disaster management, terrestrial monitoring, military monitoring, healthcare, cyber physical infrastructure systems, SCADA etc. that are the sources for big data streams [5], [19], [20]. Some applications, including terrestrial monitoring and disaster management, need data integrity so that the system has high confidence in the detected events from stream data processing; confidentiality is not that important in such applications [4], [6], [8].

II. LITERATURE REVIEW

In [5], various user authentication schemes designed for WSNs were reviewed. In this survey, the WSNs in unattended environments are targeted and secure and dependable user authentication mechanisms are analyzed for future research proposals, challenges, and applications. In [6], a study on an adaptive security design for malicious node detection in cluster-based sensor networks was presented. In this study, adaptive security modules are analyzed to improve a secure communication in cluster-based WSNs by protecting them from external malicious nodes using various authentication schemes.

In [7], a cluster-based node authentication scheme was pro- posed for WMSNs. In this plan, clusters are framed utilizing a lightweight mutual authentication mechanism among MSNs and CHs. In [8], a geographic secured routing mechanism was used to authenticate nodes and messages in WMSNs. In this mechanism, a Protected Hash Algorithm v3 (SHA-3) is utilized to limit computational multifaceted nature for authentication purposes. In [9], a user friendly hybrid authentication scheme was proposed for WSNs. In this scheme, mutual authentication and key understanding plans are consolidated utilizing tumultuous maps to limit computational multifaceted nature and ensure client anonymity and mystery passwords. In [10], a compressive sensing based approach for a secure data collection in WSNs was proposed. In this approach, asymmetric semihomomorphic encryption is used to ensure the privacy during data collection, and a sparse compressive matrix is used to reduce computational costs.

In 2005, Stonebraker et al. [11] initially highlighted the eight requirements of real time stream processing which makes stream processing research more challenging and different to batch processing. In 2009, Nehme et al. [12] proposed spotlight architecture to highlight the need for security in data streams and differentiate the security requirements of data (called *data security punctuations*) and query side security policies (called *query security punctuations*). There are a large number of security solutions proposed in the literature to protect data confidentiality and integrity by applying asymmetric and symmetric cryptography solutions [12], [13], [14], [15].In this section, we describe relevant work related to our research under the following three areas: stream processing, data stream security, and security solutions for data confidentiality and integrity.

The main contributions of the paper can be summarized as follows:

- We have developed and designed a novel Secured and Authorized Selective Encryption method (SEEN) to secure and maintain confidentiality of big sensing data streams according to different data sensitivity levels. Our method is based on common shared keys and is initialized and updated by a DSM without requiring retransmission.
- Our proposed model adopts different keys for the three levels of data confidentiality (i.e., no confidentiality, partial confidentiality and strong confidentiality) based on the data sensitivity levels. This model ensures the end-to-end security by protecting data from source device to cloud processing layer.

III. SECURED AND AUTHORIZED SEEN PROTOCOL

In this section, we explain our proposed Secured and Authorized SEEN Protocol for mobile multimedia data collection. A block diagram depicts the operational mechanism of our proposed Secured and Authorized SEEN Protocol is shown in Figure 1. Our proposed Secured and Authorized SEEN Protocol is divided in two layers, i.e., device and mobile sink layers. Clusters are formed at the device layer using a lightweight handshaking mechanism while data is collected from CHs through mobile sinks at the mobile sink layer.

3.1 Secure Cluster Formation

This phase is divided in two sub-phases, i.e., an election of CHs and a mutual authentication between MSNs and CHs. The CH election is a token-based approach which enables MSNs with highest energy levels to become the CHs. Remaining nodes joins CHs to form clusters and forward data to the elected CHs.

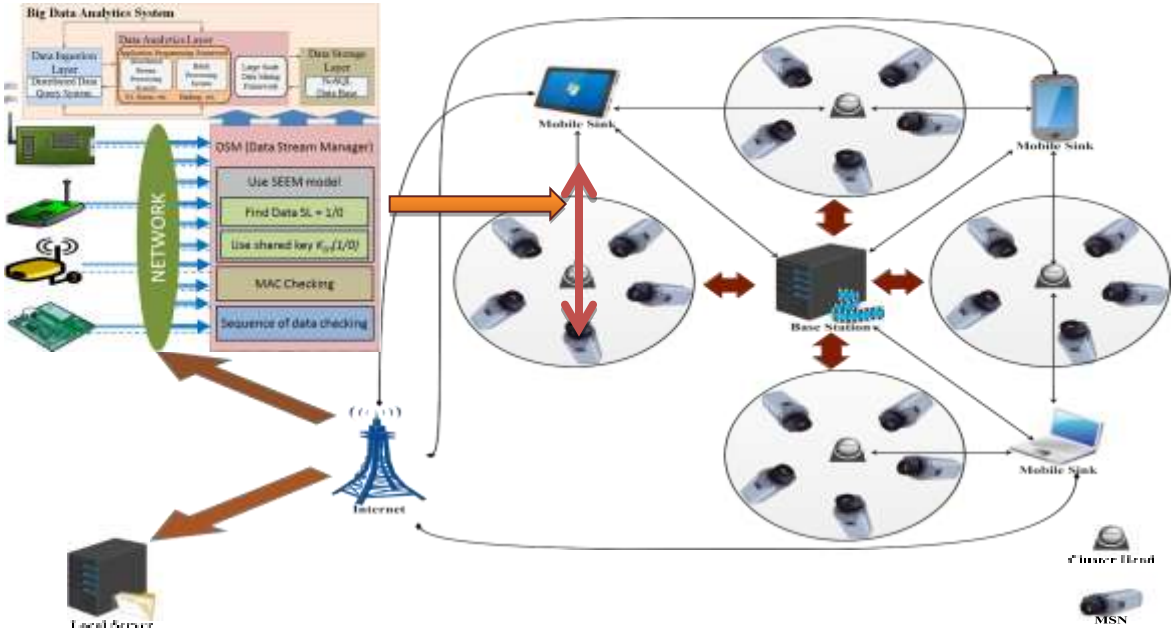

Figure 1: Operational mechanism of Secured and Authorized Selective Encryption (SEEN) protocol

3.2 CH Election

In our proposed secured and authorized SEEN protocol, a WMSN consists of 500 randomly deployed MSNs. The MSNs with minimum energy levels become members in a cluster while remaining nodes become CHs. The elected CHs are responsible for authentication of member nodes and receiving and forwarding of data to a BS. Each MSN possesses a *Z* and a t. The t is used to exchange control packets i.e., nomination packets (i.e., *p*) and acknowledgment packets (i.e., *ACK*), between the CHs and BS. On the other hand, the *Z* is used to perform mutual authentication inside clusters between the MSNs and CHs.

In the beginning of each simulation round, each a_i sends a p_i to BS. Each p_i contains a's ID (i.e., *i*), secret (i.e., Z_i), and residual energy (i.e., a_e). Upon receiving a p_i, the BS extracts *i* and a_e from *p* and computes an average energy threshold (i.e., *E*) by using the following equation.

$$E = \frac{\sum_{i=1}^{N} a_e}{N}. \quad (1)$$

where *N* represents the total number of MSNs in the WMSN. An a_i having energy equal or greater than *E* is considered eligible to be elected as a CH. An exception may occur at the start of each simulation round when all MSNs have almost the same energy levels. In this case, the election of a CH is based on minor differences in energy values and can go up to four decimal digits. In upcoming simulation rounds, it is possible to have more than one nominee for a CH role. In this case, a CH is selected based on the following rules.a_e should be equal to or greater than *E*. Node a_i is not elected as a CH in past simulation rounds.
In the case of same energy levels, nominees in the same geographical location are evaluated based on their previous history of the election.

3.3 Mutual Authentication

In this paper, we propose a Secured and Authorized Selective Encryption method for big data stream which is furnished with key renewability and makes a tradeoff among security, performance and resource utilization. This security method's salient features are as follows:

- Efficient key broadcasting without retransmission;
- Ability to recover the lost keys with a proper detection;
- Seamless key refreshment without interrupting the data streams; and
- Maintain the data confidentiality based on the data sensitivity level.

3.3.1 Initial System Setup

We follow the symmetric key method for the initial system setup because of the limited resource availability at the source sensors [11] as shown in Figure 2. In symmetric key encryption, hashing function need 5.9 mJ and encryption techniques 1.62 mJ whereas in an asymmetric key, RSA-1024 needs 304 mJ to sign and 11.9 mJ to verify and ECDSA-160 needs 22.82 mJ to sign and 45 mJ for verification [11].

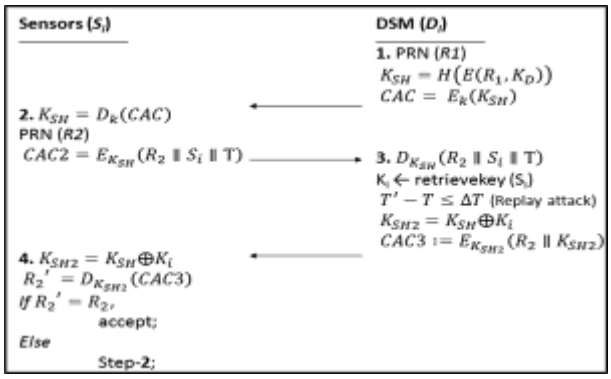

Figure 2.: Initial authentication methods with 4 step process

3.3.2 Re-Keying

After this initial key setup phase, the DSM shares the shared secret key with sensors for encryption. For the rekeying process, we follow a LiSP protocol [12] and modify it to make SEEN more data centric instead of communication centric. SEEN uses a key server (KS) at the DSM, that manages the security keys for both strong and weak encryption. We use 128 bit symmetric shared key for strong encryption and 64-bit symmetric key for weak encryption. Shared keys from KS are always chosen to perform the rekeying operation. Along with the shared key, individual sensors are able to perform the hash function as shown in Figure 3.

In order to make the system more secure, the shared key distribution for rekeying must be secure and fault tolerant; where "secure" means to maintain the confidentiality and authenticity and "fault tolerant" implies the capacity to restore the lost shared key (K_{SH}). In our SEEN method, we always use two kinds of control packets i.e., *UpdateKey* and *RequestKey*. *UpdateKey* is for periodically updating the shared key used by DSM, whereas *RequestKey* is used by sensors when they missed the shared key during the rekeying process.

Algorithm 1. Rekeying process

t – time to rekeying
t' – time to DSM starting the shared key distribution
dt – small time before t expires

1. At time t': the DSM broadcasts (*UpdateKey*)
   UpdateKey ❯ E_{KSH}ôK_{SH} ô1Þk K_{SH} ô0ÞÞ
2. Sensors use the current shared key (E_{KSH}) to get the next shared key
   D_{KSH}ôK_{SH}ô1Þ k K_{SH} ô0ÞÞ
3. At time dt: If any sensor does not have the next shared key Sensors unicasts to DSM (*RequestKey*)
   RequestKey ❯ E_{KSH}ôS_i k t_iÞ
4. After authentication, the DSM unicasts (*UpdateKey*)
   UpdateKey ❯E_{KSH}ôK_{SH}ô1ÞkK_{SH}ô0ÞÞ

Figure 3: Algorithm for Re-keying process

3.3.3 New Node Authentication

Joining new nodes to the network is a common property of sensor networks. We assume that the source node is initialized by the DSM during the initial deployment [14]. In such cases, source sensors always start the process by authenticating with the DSM to get the current shared key. Sensors use a control packet (i.e., *InitKey*) to start the process. *InitKey* contains the source ID encrypted with the initially deployed secret key i.e., E_kôS_jÞ. Once the DSM receives the control packet, it checks its authenticity. If the DSM succeeds in the authentication process, then it follows the Initial key setup (from Figure 1) phase to share the current shared key. The DSM uses the current shared key (K_{SH}) instead of generating a new key i.e., K_{SH}

¼ fHðEðR_1; K_DÞþg. At the final stage of sharing the shared key, the DSM shares the keys along with a time stamp (t_i) to source sensors (E_{KSH2} ðR_2 k K_{SH2}ðþ k t_iÞ). For the robust clock skew and shared information details, the source sensor can get the information from its neighbors [12].

3.3.4 Reconfiguration

The DSM will configure the shared key at the time of the next rekeying process, if (1) any of the source sensors have been compromised; (2) any of the shared keys have been revealed; (3) a source node has overtly requested the shared key; or (4) a source has joined to participate in the data stream. The actions required for the issues highlighted above are summarized as follows:

- DSM withdraws the compromised nodes closed previously. This may expose all earlier shared keys.
- DSM computes new shared keys for both strong and weak encryption and unicasts with control packets.
- DSM answers to the mentioning source with current configuration.
- DSM follows the cauthentication procedure, and if fruitful, DSM reacts to the source by introducing an InitKey control packet.

3.3.5 Encryption/Decryption

The above defined process makes both shared keys (K_{SH}ð1Þ k K_{SH}ðþ) available at sensors. Note that K_{SH}ð1Þ$*is always* used for strong encryption, whereas K_{SH}ðþ *is* always used for weak encryption. Each data block generated at sensors is a combination of two different parts. The first part is for integrity checking and maintaining the confidentiality level, whereas the other part is for the source authentication (i.e.,A_D ¼ E_{KSH}ð1þ ðS_i k $\frac{1}{0}$ k Tþ. The authentication part is always encrypted using the strong encryption key; it contains the source ID for authentication, time stamp (T) to avoid replay attack, and a flag value 1/0;where 1 is for strong encryption of body part (highly sensitive data) or 0 for weak encryption of body part (low sensitivity data). In order to encrypt the data part of the packet, every sensor performs the XOR operation i.e., current shared keys K_{SH}ð1=0þ with its own secret key ðk_iþ, i.e., K_{SH}ð1þ$^{()}$ ¼ K_{SH}ð1Þ K_i and K_{SH}ðþ$^{()}$ ¼ K_{SH}ðþ Þ $K_i^{()}$ then it uses the newly generated key to encrypt the data packets. Shared key $E_{KSH(1)}$() is always used for strong encryption, whereas shared key $E_{KSH(0)}$() is used for weak encryption (DATA k MAC) shown in Figure 4. The above specified data block encryption is always based on the data sensitivity level and for data integrity and confidentiality.

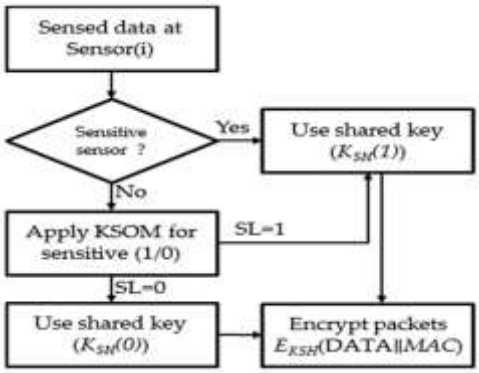

Figure 4: Method to select encryption method based on the data sensitive level

3.4 Mobile Sink Registration

For a secure data exchange between CHs and mobile sinks, there is a need to register the mobile sinks with a BS as shown in Figure 5. In real-world scenarios, the mobile sinks are always on the move. As a result, the CHs get a very short amount of time to share data with authorized mobile sinks. In the proposed protocol , each b$_k$ generates an m$_k$ and some random integers, i.e., x and z. The generated m$_k$ and random integers are used to register a b$_k$ with the BS and these numbers need to be generated before forwarding a registration request. The registration process is based on two hashing functions, i.e., H_1 and H_2, for mapping and secure hashing purposes, respectively [22]. The first hashing function, i.e., H_1, is used to perform a map to point hashing operation and the second hashing function, i.e., H_2, is used to perform one-way secure

hashing. For both hashing operations, the range of values is between 0 and 1.

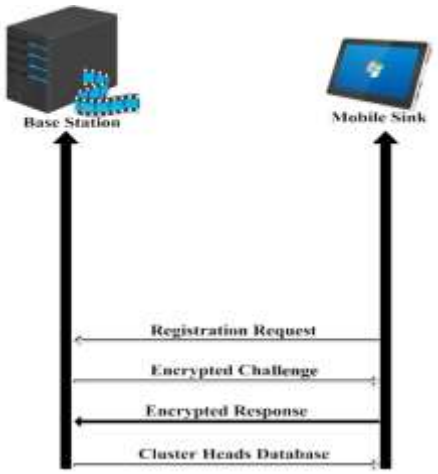

Figure 5: Registration of mobile sink with base station

After receiving the encrypted response, the BS verifies the identity of the mobile. If the equation holds, the mobile sink is considered authentic otherwise it is considered an intruder node. After verifying the identity of a mobile sink, the BS performs three operations, 1) sharing of registration certificate and time-stamp of verified mobile sink with all CHs, 2) forwarding database of elected CHs to verified mobile sinks, and 3) forwarding location information of nearby CHs to verified mobile sinks to complete the registration process.

Once the mobile sink broadcasts a request for a data collection, the nearby CHs reply back with capacity allocation requests based on their low and high priority data. The mobile sink first processes high priority data requests and assigns high capacities to the requesting CHs. Once the capacity is assigned, the CHs immediately start transmitting data. If there is any capacity left at the mobile sink, it is evenly assigned to the CHs with low priority data requests. The capacity allocation process is a continuous process and stops when no more capacity is left.

IV. EXPERIMENT AND RESULT

In order to evaluate the security strength and efficiency of the SEEN security method under the above specified adverse situations, we experimented in multiple simulation environments. The experiment was conducted using the in- house simulators on an Intel (R) Core (TM) i5-6300 CPU @ 2.40 GHz 2.50 GHz CPU and 8 GB RAM running on Micro- soft Windows 7 Enterprise. We first verified the proposed security approach using Scyther [28]; second, we measured the performance of the approach using JCE (Java Cryptographic Environment) [29]; third, we computed the required buffer size to process our proposed approach using MatLab [30] to measure the efficiency of our method; finally, we used COOJA simulator in Contiki OS [31] to get the network performance of SEEN.

4.1 Security Verification

The SEEN security protocol is simulated in the Scyther simulation environment by using the underlying Security Protocol Description Language (.spdl). Scyther is an automatic security protocol verification tool that can be used to check the correctness of the security protocols. As per the Scyther model, we defined the roles of S and D, where S is a sensing device and D is the receiver (i.e., DSM). In this scenario, S and D have all information for encryption/decryption that is initialized in the system setup and rekeying phase. In this simulation environment, S sends the encrypted data packets to D for security verification. We introduced three types of attacks. First, an attacker changes the data packet while it is in the network. In the second, an adversary steals the property of source (S) and forwards the data packets to D pretending to be S. In the third, an adversary gets the data block to analyze and tries to read the data and replay the data packets.

4.2 Performance Comparison

We used JCE (Java Cryptographic Environment) to experiment on and evaluate the performance of the SEEN method. JCE is the standard extension to the Java platform that provides an implementation context for cryptographic methods. The experiment is based on the features of the JCE in 64 bit Java virtual machine version 1.6. The experiment outcomes for security verification are shown in Figure 6. We

compare the performance of SEEN security with advanced encryption standard (AES-128, AES-192), LSec and our previously proposed model for big sensing data streams i.e., DPBSV and DLSeF [19], [20].

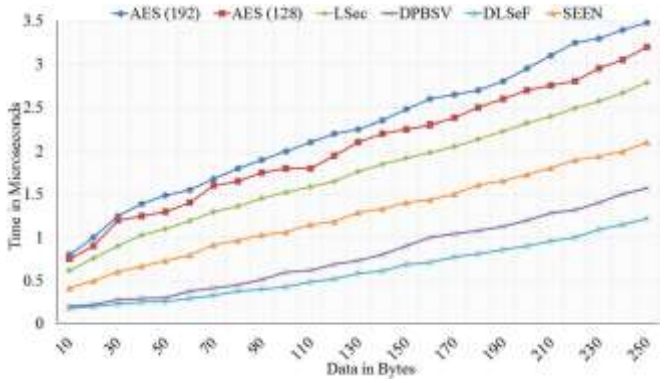

Figure 6: Performance comparison SEEN method with AES-128, AES-192, LSec, DPBSV, and DLSeF

4.3 Network Performance

We tested our SEEN protocol using a COOJA simulator in Contiki OS to get the network performance (i.e., communication overhead and power consumption) [31]. We took the two most common types of sensor (i.e., Z1 and TmoteSky sensors) for network simulation. In this experiment, we checked the performance while computing and distributing the shared key. For network simulation, we took a random area to deploy 51 nodes (i.e., 50 sensors and 1 DSM) in COOJA simulation environment. We took initial battery power of an individual sensor node $1x10^6$ J, power consumption for transmission is 1.6W and power consumption for reception is 1.2 W. Apart from these, we follow the default properties of sensors. We assume that the size of each data packet is 30 bytes, nonce 23 bits, and secret key of 64/128 bits and token 4 bytes for the simulation[4].

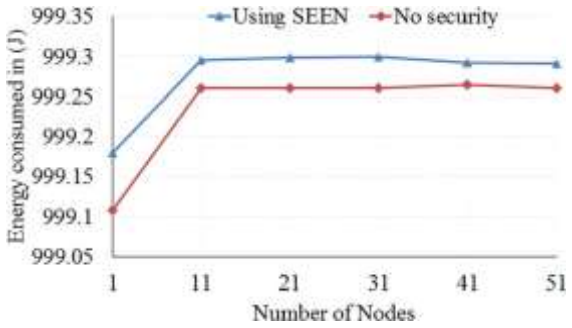

Figure 7: Network Performance

For every connection, SEEN exchanges control packets for source/DSM authentication and shared key distributions based on the above specified packet sizes. This is an acceptable tradeoff between energy and security for the sensor node. The simulation results of Network Performance are shown in Figure 7. The Secured and Authorized SEEN protocol required extra energy for the network authentication but its difference is very low. The energy consumption by using this protocol remains the same even by increasing the network size. We simulated the scenario using 50 nodes in 10 nodes interval as shows in Figure 7.

V. CONCLUSION

In this paper, we have proposed a secure data collection scheme, called Secured and Authorized SEEN Protocol. It has been designed to collect multimedia data using mobile sinks in a wireless multimedia sensor network. The proposed Secured and Authorized SEEN Protocol is based on a clustering concept. A selective encryption mechanism is applied to authenticate both MSNs and CHs. Once authenticated, the MSNs forward captured data to elected CHs, and then it is the responsibility of CHs to share the collected data with nearby verified mobile sinks. The mobile sink verification involves their registration with the base station and authentication with the CHs. The simulation results have shown that our proposed Secured and Authorized SEEN Protocol performs better in terms of security verification, performance comparison and network performance when compared with other established state of the art approaches. Regarding future work, we shall use the simulation results produced in this paper as a base to perform further enhancements in our proposed Secured and Authorized SEEN Protocol to support mobile

MSNs and sinks with random speeds.

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surv. Tut.,* vol. 17, no. 4, pp. 2347–2376, Oct.–Dec. 2015.

[2] S. A. Alvi, B. Afzal, G. A. Shah, L. Atzori, and W. Mahmood, "Internet of multimedia things: Vision and challenges," *Ad Hoc Netw.,* vol. 33, pp. 87–111, 2015.

[3] H-S. Lim, Y-S. Moon and E. Bertino, "Provenance-based trustworthiness assessment in sensor networks," in *Proc. 7th Int. Workshop Data Manage. Sensor Netw.,* 2010, pp. 2–7.

[4] S. Sultana, G. Ghinita, E. Bertino and M. Shehab, "A lightweight secure provenance scheme for wireless sensor networks." in *Proc. 18th Int. Conf. Parallel Distrib. Syst.,* 2012, pp. 101–108.

[5] S. Kumari, M. K. Khan, and M. Atiquzzaman, "User authentication schemes for wireless sensor networks: A review," *Ad Hoc Netw.,* vol. 27, pp. 159–194, 2015.

[6] M. A. Jan, P. Nanda, X. He, and R. P. Liu, "A sybil attack detection scheme for a forest wildfire monitoring application," *Future Generation Comput. Syst.,* vol. 80, pp. 613–626, 2018.

[7] M. Usman, M. A. Jan, X. He, and P. Nanda, "Data sharing in secure multimedia wireless sensor networks," in *Proc. IEEE Trust-com/BigDataSE/ISPA,* 2016, pp. 590–597.

[8] B. P. Laxmi and A. Chilambuchelvan, "GSR: Geographic secured routing using SHA-3 algorithm for node and message authentica- tion in wireless sensor networks," *Future Generation Comput. Syst.,* vol. 76, pp. 98–105, 2017.

[9] S. Kumari, X. Li, F. Wu, A. K. Das, H. Arshad, and M. K. Khan, "A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps," *Future Genera- tion Comput. Syst.,* vol. 63, pp. 56–75, 2016.

[10] P. Zhang, S. Wang, K. Guo, and J. Wang, "A secure data collection scheme based on compressive sensing in wireless sensor networks," *Ad Hoc Netw.,* vol. 70, p. 73–84, 2018.

[11] M. Stonebraker, U. Çetintemel, and S. Zdonik, "The 8 requirements of real-time stream processing," *ACM SIGMOD Rec.,* vol. 34, no. 4, pp. 42–47, 2005.

[12] R. V. Nehme, H-S. Lim, E. Bertino, and E. A. Rundensteiner, "StreamShield: A stream-centric approach towards security and privacy in data stream environments." in *Proc. ACM SIGMOD Int. Conf. Manage. Data,* 2009, pp. 1027–1030.

[13] S. Sultana, G. Ghinita, E. Bertino and M. Shehab, "A lightweight secure provenance scheme for wireless sensor networks." in *Proc. 18th Int. Conf. Parallel Distrib. Syst.,* 2012, pp. 101–108.

[14] R. A. Shaikh, S. Lee, M. AU Khan and Y. J. Song, "LSec: Lightweight security protocol for distributed wireless sensor network." in *Proc. IFIP Int. Conf. Personal Wireless Commun.,* 2006, pp. 367–377.

[15] G. Selimis, et al., "A lightweight security scheme for wireless body area networks: design, energy evaluation and proposed micropro-cessor design." *J. Med. Syst.,* vol. 35, no. 5, pp. 1289–1298, 2011.

[16] G. Selimis, et al., "Evaluation of 90 nm 6 T-SRAM as physical unclonable function for secure key generation in wireless sensor nodes," in *Proc. IEEE IEEE Int. Symp. Circuits Syst.,* 2011, pp. 567– 570.

[17] Muhammad Usman, Mian Ahmad Jan, Xiangjian He and Jinjun Chen, "A Mobile Multimedia Data Collection Scheme for Secured Wireless Multimedia Sensor Networks" IEEE Transactions On Network Science And Engineering, VOL. 7, NO. 1, 2020

[18] Deepak Puthal , Xindong Wu , Nepal Surya , Rajiv Ranjan , and Jinjun Chen," SEEN: A Selective Encryption Method to Ensure Confidentiality for Big Sensing Data Streams", IEEE Transactions On Big Data, VOL. 5, NO. 3, 2019

[19] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "DLSeF: A dynamic key length based efficient real-time security verification model for big data stream," *ACM Trans. Embedded Comput. Syst.,* vol. 16, no. 2, 2017, Art. no. 51.

[20] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "A synchronized shared key generation method for maintaining end-to-end security of big data streams," in *Proc. onf. Syst. Sci.,* 2017, pp. 6011–6020.