A Mobile-device-based Event Reporting Scheme for WSNs

Chin-Ling Chen
Department of Computer Science and Information Engineering
Chaoyang University of Technology
Taichung 41349, Taiwan, ROC
e-mail: clc@mail.cyut.eu.tw

Yong-Yuan Deng
Department of Information Management
Chaoyang University of Technology
Taichung 41349, Taiwan, ROC
e-mail: allen.nubi@gmail.com

De-Kui Li
Department of Logistics Management
Wuhan Technology and Business University, Wuhan, China
e-mail: jerryinkorea@gmail.com

***Abstract*—In recent years, an increasing number of researchers have become involved in wireless sensor networks (WSNs) and cloud computing. However, integrating WSN and cloud computing technology to monitor the environment is still an open issue. In this paper, we propose a mobile-device-based environmental monitor based on SaaS (Software as a Service) of cloud computing architecture. The proposed model is a mobile-device-based event reporting scheme. In an environmental monitoring system since there are some events (such as structural damage to dams and bridges) that cannot be reported automatically, we propose using a mobile device to report such events. The proposed scheme can defend against Denial of Service (DoS) and man-in-the-middle attacks, achieve mutual authentication, ensure data security and address network security issues.**

Keywords- wireless sensor network; cloud computing; mobile device security; sensor node·sink

I. INTRODUCTION

In recent years, cloud computing and research concerning wireless sensor networks (WSN) have become increasingly relevant. Integration of cloud computing and wireless sensor networks has become an important consideration. Cloud computing includes application service provision (ASP); it incorporates some features that it can reliably support. There are many enterprises which define cloud modes, including Google, IBM [1] and Amazon [2]; they provide different services by themselves.

In this paper, we propose two event reporting models. One is a two-tier data aggregation for grid-based WSNs via a SaaS cloud computing architecture; the other is a mobile-device-based event reporting scheme using SaaS cloud computing architecture.

The first WSN environmental monitor is an automatic model which reports event data (such as flood and fire events) to a spare two-tier data aggregation of sink and then sends the event message to a cloud server. Due to advances in wireless and IC process technology, a WSN was applied to replace traditional network technologies [5, 6, 7, 8]. Small, low-cost sensors have recently become technically and economically feasible [9]. WSN has been effectively applied to both military and civil applications, including: intrusion detection, weather monitoring, security and tactical surveillance, and distributed computing. In 2002, Ye et al. [10] proposed a two-tier data dissemination model (the TTDD scheme); here the source announces the event using grid dissemination nodes once the source detects that an event has occurred. The dissemination nodes transmit packets when the sink queries the event. However, the path from the source to the sink is not optimal. In addition, in 2004, a coordination-based data dissemination protocol (the CODE scheme) [11] was proposed for a grid-based sensor network. An efficient data dissemination path was established based upon the location of event and grid IDs. If the sink moves from the original grid, a new one needs to be reconstructed. However, CODE did not address the routing problem when obstacles or voids were encountered in a sensor field. In 2010, Hung et al. [12] discussed about secure WSN-integrated cloud computing. When a data is sent to the cloud server from WSN, it has to be classified into event data through cloud gateway. The filtering module of the cloud gateway filtered redundant and noise data to reduce communication overhead before sending to the cloud. Our proposed protocol is the first which not only combines data dissemination and data aggregation, but also solves the problem of tracking diffused event and data aggregation in the sensor field which combines the cloud computing application with WSN. We have also used a NS2 simulation tool to simulate our scheme, and proved that it outperforms both TTDD and CODE.

The second model conducts mobile-device-based event reporting via SaaS cloud computing architecture. In an environment monitoring system, there are some events (such as the structural damage of dams and bridges) that cannot be reported automatically. With this in mind, we propose a manual reporting scheme based on the use of a mobile device. As a SaaS model, the data is stored in the SaaS provider's data center; if the SaaS is constructed based on public cloud computing architecture, it may involve many security issues [13, 14]. As a result, the proposed scheme uses a one-way hash function [15, 16] to mask the secret parameters. If the investigator wants to send the message to a cloud server, the messages must be encrypted by session key. In 2010, Subashini et al. [17] proposed a survey of cloud server security and identified the following security issues for SaaS:

QSHINE 2015, August 19-20, Taipei, Taiwan

DOI 10.4108/eai.19-8-2015.2260697

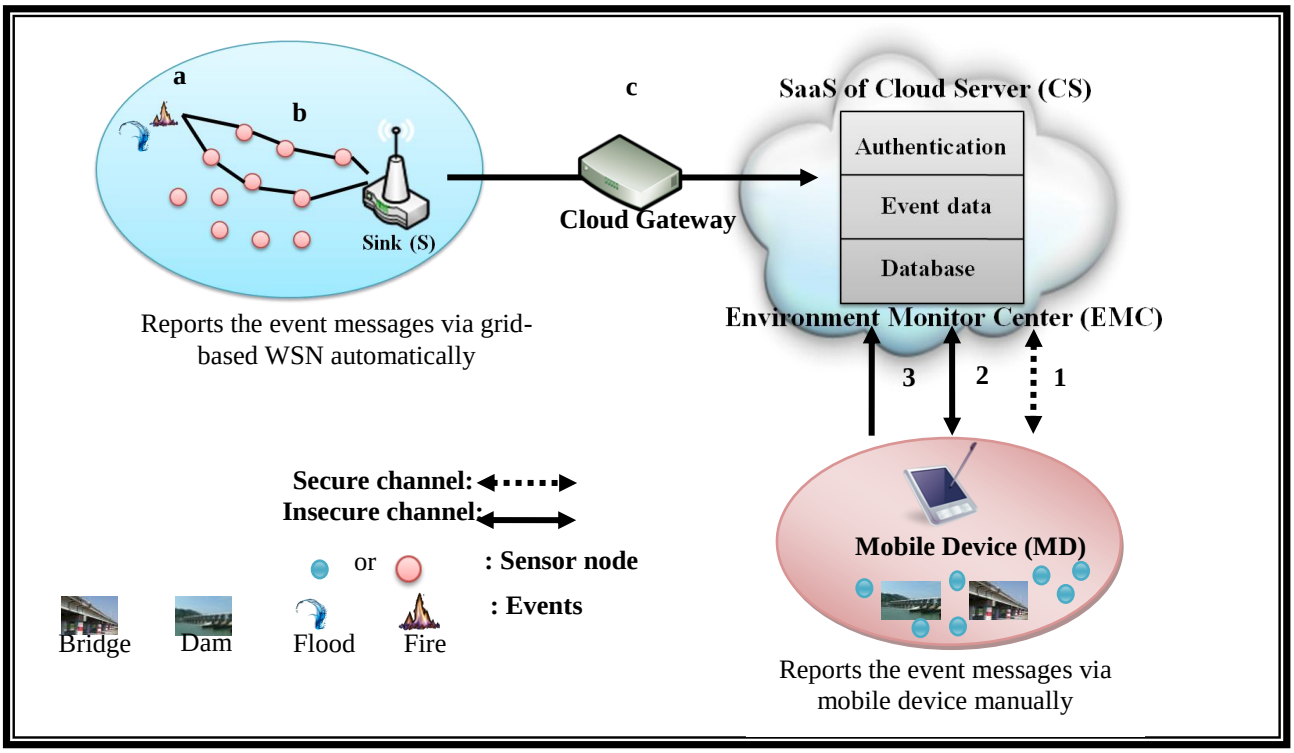

Figure 1. The framework of our scheme

(1) Mutual authentication: Authentication is important for both server and user; the cloud server can use mutual authentication to identify the user's identity.
(2) Denial of Service (DoS) attacks: Cloud computing is a public environment for each user. If an attacker finds the cloud server's weakness, he/she can aim at the weakness by sending garbage packets to the cloud server again and again. This goes on until the cloud server collapses and can no longer provide normal service.
(3) Man-in-the-middle attacks: This attack is carried out when an attacker employs two roles. The attackers are placed in the communication path where they can intercept and modify the message.
(4) Data security: In the SaaS model, a user's data is stored in the SaaS provider rather than at the user end. Therefore, the SaaS provider must use additional techniques to enhance data security between the user and the cloud server.
(5) Network security: With the SaaS model, the SaaS application can access sensitive data from the user and store them at the SaaS provider end. Therefore, all data flow over the network must be protected.

The rest of this paper is organized as follows. In Section 2, we describe the architecture of our scheme. Section 3 presents a mobile-device-based event reporting scheme. In Section 4, the security analysis of a mobile-device-based event reporting scheme is presented. Finally, conclusions are discussed in Section 5.

II. THE ARCHITECTURE OF OUR SCHEME

In an environment monitoring area, numerous event messages must be reported to the monitoring center. The center then stores the event messages into the cloud sever which has been constructed by the environment monitoring center. In this section, we will outline how to automatically report the data aggregation using the WSN model, and how to report the event messages manually to the cloud server using a mobile device. Figure 1 illustrates the framework of our scheme; it is divided into two models for reporting data aggregation.

A. Two Models of Reporting Data Aggregation

Based upon a SaaS cloud computing architecture, we propose two event reporting models. In SaaS models, the user does not need to install any software in the computer or mobile device; if the device is authorized to access the network, the user can access the service of SaaS. In 2011, Subashini et al. [15] proposed security requirements for the SaaS stack. Because user data is stored at the SaaS provider end, the cloud provider has to verify the user's identity and satisfy the security requirements.
(1) Reporting Event Messages Automatically via a Grid-Based WSN model

When a significant event (such as a fire or flood event) happens, a sensor node within the region will automatically detect this event. If a neighboring node detects the event, it will forward this packet and store the message in an events table. We describe the event process below.

Step 1: When an event of interest occurs, a sensor node within the region will detect this event.
Step 2: The sensor node provides a spare two-tier data aggregation and sends the event data to the sink.
Step 3: After the sink receives the event data, it can be

classified out noise data by cloud gateway before sending to the cloud architecture of environment monitor center [8].

(2) Manually Reporting Event Messages via a Mobile Device

Since some events within a region, such as structural damage to dams and bridges, cannot be detected and automatically transmitted using WSN, we propose using a mobile-device-based manual reporting model. The steps are described as follows.

Step 1: MD $\leftrightarrow$ CS: The investigator can take a mobile device (MD) and legally register it through secure channels.

Step 2: MD $\leftrightarrow$ CS: Before an investigator can send event data to the cloud server, he/she must login as a legal user of the cloud server, and then perform mutual authentication between the mobile device and the cloud server.

Step 3: MD $\rightarrow$ CS: When an investigator patrolling the area of interest discovers an event, the event message must be sent in real time to the environment monitoring center via SaaS.

As in 2010, we proposed a spare two-tier data aggregation for grid-based wireless sensor networks[18]. In that scheme, it has achieved the reporting event messages automatically via a grid-based WSN model. Therefore, we will aim to the manually reporting event messages via a mobile device in the following sections.

II. A MOBILE-DEVICE-BASED EVENT REPORTING SCHEME

In this section, we propose an mobile-device-based event reporting scheme. When an investigator discovers some events (such as the structural damage of dams or bridges) which can not be detected via automatic report as described in section 2, the investigator can report events manually using a mobile device. There are three phases: the registration phase, login and authentication phase, and the communication phase. The notation and the steps involved in each phase are as follows:

A. Notation

In this section, we denote the parameters of our mobile-device-based scheme as follows.

MD	: the investigator's mobile device
CS	: the cloud server of the environment monitoring center
ID_X	: the identity of X
PW	: the password
r	: the random value of the investigator's mobile device
x	: the master secret key of the cloud server
y	: the short term secret parameter of the cloud server
SK	: the common session key
N_X	: the nonce value of X
m	: the event message which is discovered by an investigator
$\|\|$	: concatenation operation
$\oplus$	: the XOR operation
$E_k(.)/D_k(.)$	: the symmetric encryption / decryption function with secret key k
$h(.)$	: the one-way hash function

B. Registration Phase

In the registration phase, the MD has to register with the cloud server which has been constructed by the environmental monitoring center. Before the investigator takes the mobile device to the environment monitoring region, he/she has to register to be a legal identity with CS as follows.

Step 1: The investigator enters the identity ID_{MD} and passwords PW, the investigator's mobile device generates the random value r, and then sends $(ID_{MD}, h(PW) \oplus r)$ to the cloud server via a secure channel.

Step 2: When the cloud server receives $(ID_{MD}, h(PW) \oplus r)$, it generates the nonce N_{CS} for the MD, and computes P, Q, R and V.

$$P = h(ID_{MD} \| h(PW) \oplus r) \quad (1)$$

$$Q = (ID_{MD} \| N_{CS} \oplus (h(PW) \oplus r) \oplus h(x)) \quad (2)$$

$$R = (h(x \| N_{CS}) \| y) \oplus P \quad (3)$$

$$V = h(ID_{MD} \| (h(PW) \oplus r) \| h(x \| N_{CS})) \quad (4)$$

Then, the cloud server sends the related parameters (Q, R, V, $h(.)$) to the mobile device using a secure channel.

C. Login and Authentication Phase

When an investigator wants to patrol the environment, he/she has to login to the cloud server a first time. In this phase, our scheme not only verifies the owner's identity, but also checks to verify if the cloud server is legal.

When the investigator wants to login to the cloud server, he/she has to perform the following procedures:

Step 1: If the investigator wants to login to the cloud server, he/she first enters his/her own identity ID'_{MD} and password PW'.

After the investigator inputs the identity and password, the MD can obtain $h(x \| N_{CS})'$ and secret parameter y as follows.

$$(h(x \| N_{CS})' \| y') = R \oplus h(ID'_{MD} \| h(PW') \oplus r) \quad (5)$$

When the MD obtains $h(x \| N_{CS})'$, the MD compares $h(ID'_{MD} \| (h(PW') \oplus r) \| h(x \| N_{CS})')$ and V to check whether the investigator is legal as follows.

$$h(ID'_{MD} \| (h(PW') \oplus r) \| h(x \| N_{CS})') \stackrel{?}{=} V \quad (6)$$

If Eq. (6) holds, the MD generates nonce value N_{MD}. Since ID'_{MD}, $h(PW)' \oplus r$, N_{MD} and Q are included in the C_1

$$C_1 = (ID'_{MD} \| (h(PW') \oplus r) \| N_{MD} \| Q) \quad (7)$$

then MD computes the session key k to encrypt C_1 as follows.

$$k = h(x \| N_{CS}) \quad (8)$$

$$C_2 = E_k(C_1) \quad (9)$$

$$E = h(ID'_{MD} \| h(x \| N_{CS})') \quad (10)$$

Afterward, MD sends the (E, C_2) to CS; otherwise, the login request is rejected.

Step 2: After receiving the message (E, C_2), the CS uses his master secret key x and generated N_{CS}, combine with different ID_{MD}, the CS uses hash operation to compute $E' = h(ID''_{MD} \| h(x \| N_{CS})'')$, and then compares with E as follows.

$$E' \stackrel{?}{=} E \quad (10)$$

If Eq. (10) holds, the CS also gets the session key k as follows.

$$k = h(x \| N_{CS}) \quad (11)$$

the CS decrypts the C_2 by session key k as follows.

$$(ID'_{MD} \| (h(PW') \oplus r) \| N_{MD} \| Q) = D_k(C_2) \quad (12)$$

and obtains the ID_{MD} and N_{CS} from Q.

$$(ID_{MD} \| N_{CS}) = Q \oplus (h(PW') \oplus r) \oplus h(x) \quad (13)$$

and then compares ID_{MD} with ID'_{MD} to check MD's legality as follows.

$$ID'_{MD} \stackrel{?}{=} ID_{MD} \quad (14)$$

If Eq. (14) holds, it means the MD had been registered. Otherwise, the login request is rejected. Next, the CS updates secret parameter y_{new} for next communication, and uses $h(x \| N_{CS})$ and N'_{MD} to make a session key SK:

$$SK = h(h(x \| N_{CS}) \| N'_{MD}) \quad (15)$$

and then the CS uses the secret parameter y to compute symmetric encryption as follows:

$$C_3 = E_y(SK \| y_{new}) \quad (16)$$

The CS also updates new nonce value N_{CSnew} for next communication, and the CS uses the secret parameter k to compute symmetric encryption as follows:

$$C_4 = E_k(h(x \| N_{CSnew}) \| (N_{CSnew} \oplus h(x))) \quad (17)$$

Then, the CS sends the message (C_3, C_4) to the MD.

Step 3: When receiving (C_3, C_4) from the CS, the mobile user (investigator) has to use the y (y' of Eq. (5)) to decrypt the message C_3 by secret parameter y, and obtain SK, and new secret parameter y_{new} as follows:

$$(SK' \| y_{new}) = D_y(C_3) \quad (18)$$

Afterward, the investigator obtain the SK and y_{new}, and checks the SK whether correct as follows:

$$h(h(x \| N_{CS})' \| N_{MD}) \stackrel{?}{=} SK \quad (19)$$

If Eq. (18) holds, the MD can use the session key SK to encrypt and protect the event data; otherwise, the MD rejects to send the event data to the CS.

Then, the mobile user (investigator) decrypts the message C_4 by session key k, and obtain $h(x \| N_{CSnew})$ and $(N_{CSnew} \oplus h(x))$ as follows:

$$h(x \| N_{CSnew}) \| (N_{CSnew} \oplus h(x)) = D_k(C_4) \quad (20)$$

The MD also updates Q, R and V to Q_{new}, R_{new} and V_{new} for next communication as follows:

$$Q_{new} = Q \oplus (N_{CS} \oplus h(x)) \oplus (N_{CSnew} \oplus h(x)) \quad (21)$$

$$R_{new} = (h(x \| N_{CSnew}) \| y_{new}) \oplus P \quad (22)$$

$$V_{new} = h(ID_{MD} \| (h(PW) \oplus r) \| h(x \| N_{CSnew})) \quad (23)$$

D. Communication Phase

In the communication phase, when the MD wants to send the event message to the CS, it has to protect the event message by session key. When the investigator wants to send the event message m to the CS, he/she has to compute the equation as follows.

Step 1: The MD encrypts the event message m by symmetric encryption with session key SK as follows:

$$C_5 = E_{SK}(m) \quad (24)$$

And then the MD sends the message C_4 to the CS.

Step 2: After receiving the message C_5, the CS decrypts the message C_5:

$$m = D_{SK}(C_5) \quad (25)$$

When the CS gets the message m, it can release the event data to the website and store the event data to database.

III. The Security Analysis and Simulation Result

A. Security Analysis for Manual Report Method

In this section, we use several security requirements mentioned above to analyze the proposed scheme, the proposed scheme can achieve these security requirements.

(1). Mutual authentication issue

Case 1: The cloud server authenticates the mobile device

The mobile device (MD) can check if the user is the real owner as identified by verifying the following equation:

$$h(ID'_{MD} \| (h(PW') \oplus r) \| h(x \| N_{CS})') \stackrel{?}{=} V \quad (6)$$

Also the cloud server (CS) can check the identity of mobile device by verifying the following equation:

$$ID'_{MD} \stackrel{?}{=} ID_{MD} \quad (14)$$

If the above equation holds, it means the mobile device of the investigator is a legal user.

Case 2:The mobile device authenticates the cloud server. When the MD receives verification message from cloud server, it can check the session key SK as follows:

$$h(h(x \| N_{CS})' \| N_{MD}) \stackrel{?}{=} SK' \quad (19)$$

If the above checks pass, then the MD can confirm the CS is a legal server, and use the session key SK to encrypt event data and send the event messages. Therefore, our scheme achieves mutual authentication.

(2). Denial-of-Service (DoS) attack issue

If an attacker wants to perform a DoS attack between the investigator and the cloud server (CS), it will fail. In our scheme, no matter which mobile device or cloud server is used, the correctness of the messages from Equs. (10), (14) and (19) must be checked. Both the CS and mobile user's identity are authenticated. Thus, our scheme can prevent a DoS attack.

(3). Man-in-middle attack issue

In our authentication phase, any user must verify correctness of identity and message in the course of each transmission. That is, even if the mobile device is lost, the user should be authenticated via Eq. (6).

$$h(ID'_{MD} \| (h(PW') \oplus r) \| h(x \| N_{CS})') \stackrel{?}{=} V \quad (6)$$

Only the real owner of the mobile device can pass the authentication. Moreover, the mobile device can authenticate the CS legality by Eq. (19):

$$h(h(x \| N_{CS})' \| N_{MD}) \stackrel{?}{=} SK' \quad (19)$$

Thus, our proposed scheme can defend against a man-in-middle cryptographic attack.

(4). Data security

In our manual reporting scheme, the events data m is protected by session key; we used the session key *SK* to encrypt the message m:

$$C_5 = E_{SK}(m) \quad (24)$$

and the session key *SK*:

$$SK = h(h(x \| N_{CS}) \| N_{MD}) \quad (15)$$

It is difficult to get the nonce value N_{CS}, N_{MD} and the secret master key *x*; the session key *SK* is updated after each session. Since the secret parameters (such as secret key *x*, short term security parameter *y*, the identity of the investigator ID_{MD}, mobiles user's password *PW* and nonce values N_{CS}, N_{MD} etc) are protected by a one-way hash function, it is difficult to gain access to these secret messages. Our scheme does achieve a high level of data security.

(5). Network security

In our system, the mobile device has to register with the environment monitoring center the first time it is used. In the registration phase, the mobile device will register via a secure channel (such as Wireless Transport Layer Security (WTLS) to obtain a legal identity. On the insecure channel, all the secret information is always protected by symmetric encryption and a one-way hash function. Thus, we are also able to achieve network security.

B. Simulation Results of a Spare Two-tier Aggregation of WSN

In this section, we describe the simulation and illustrate the simulation results for a spare two-tier data aggregation of WSN. As shown in Table 1, we first make some assumptions concerning the parameters of the system architecture.

We conducted a performance evaluation using NS2 (Network Simulation 2). The energy model used in NS2 requires about 0.7 Watt, 0.35 Watt, and 0.035 Watt for transmitting, receiving and idling, respectively. The initial battery status equaled 10 Joules. The mobility model was constructed according to the random waypoint model. The sensor nodes were deployed uniformly in a 500 m × 500 m field. The simulation lasted for 100 s. Each simulation was run 50 times.

TABLE 1. PARAMETERS USED IN THE SIMULATIONS.

Parameter	Values
Simulation tool	NS2
Simulation area	500 m × 500 m
Number of nodes	200 nodes
Sink mobility speed	5 m / sec
Sink mobility model	Randomwaypoint model
Radiotransmission range	100 m
Data packet size	512 bytes
Data sending rate	1 Mbps

The performance metrics used were as follows:

1. Total energy consumed: The total energy consumed by our system. It includes three energy dissipations: communication energy dissipation, computation energy dissipation and sensing energy dissipation.
2. Average delay: The delay is defined as the average time between the time sensing nodes transmit a packet and the time a sink receives the packet.
3. Total Energy Consumed: A comparison of the total energy consumed by the whole network is shown in Figure 2. Figure 2 shows the total energy consumed with a varying number of sinks, ranging from 1 to 5 sinks. We assume that the mobility speed is 15 m/s. The region of interest is displayed in 3x3 grids. The 200 sensor nodes are deployed uniformly in a field. The total energy consumption of our proposed scheme, CODE, and TTDD all increased when the number of sinks increased, but the total energy consumption of our scheme was lower than that of CODE and TTDD.

   Figure 3 shows the performance of the total consumed energy under a varying number of grids, ranging from 1 to 25 grids covering a region of interest. We assumed that the mobility speed is 15 m/s. Only one sink is in this network. The total energy consumption of our proposed scheme, CODE, and TTDD all increased as the number of grids increased, but the total energy consumption of our scheme was lower than CODE and TTDD, even when the number of grids increased. The energy of transmission will increase.

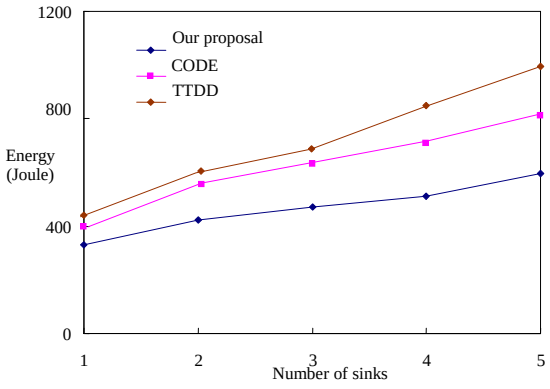

FIGURE 2. THE ENERGY CONSUMED OF DIFFERENT NUMBER OF SINKS.

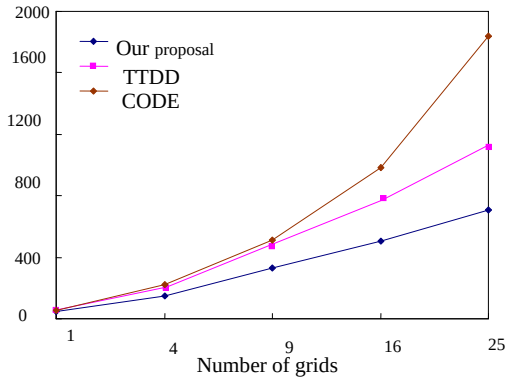

FIGURE 3. THE ENERGY CONSUMED OF DIFFERENT NUMBER OF GRIDS.

Figure 4 shows the performance of the total energy consumed under different maximum speeds of sinks, which ranged from 0 m/s to 20 m/s in the region of interest. We assumed that the number of sinks was one. The total energy consumption of our proposed scheme, CODE and TTDD increased when the maximum speed of sinks increased, but the total energy consumption of our scheme was lower than that of CODE and TTDD.

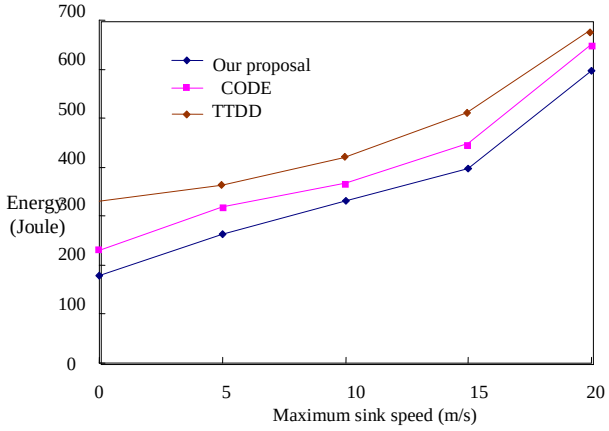

FIGURE 4. THE ENERGY CONSUMED OF DIFFERENT MAXIMUM SPEED OF SINKS.

V. CONCLUSIONS

In this paper, we proposed two different models to report events for the environment monitoring center. The first scheme is based on a WSN model and automatically reports event messages; the second scheme is based on the use of a mobile device to manually report event messages. We propose a SaaS of cloud architecture to integrate these into an all-in-one environment monitoring system.

For the automatic model, WSN provides two paths to transmit event messages in the area of interest. It can enhance the fault tolerance. By choosing the highest residual energy of the sensors, each node chooses its child node among its neighbors based on the information of the residual energy distance to the sink; t his can reduce energy consumption to extend the lifetime of the system. We used a NS2 simulation tool to simulate the proposed scheme. From the evaluation in Section 4, it would appear the contribution is our proposed protocol that expends less energy than related ones. The results show that our proposed scheme outperforms both CODE and TTDD.

With our manual model, the investigator can take a mobile device to patrol the area of interest, and report the event messages to the cloud server using the mobile device. Our scheme can not only defend against Denial of Service (DoS) attacks and man-in-the-middle attacks but also achieve mutual authentication and ensure data security and network security. Therefore, our approach is feasible.

ACKNOWLEDGMENT

This research was supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract number MOST 103-2632-E-324-001-MY3, MOST 103-2622-E-212-009-CC2 and MOST 103-2221-E-324 -023.

REFERENCES

[1] IBM point of view security and cloud computing, (Available from: http://www-07.ibm.com/systems/nz/itsolutions/security/pdf/TIW14045USEN_HR.pdf [June 11, 2013]).

[2] Amazon; cloud architectures. (Available from: http://aws.amazon.com/ [June 11, 2013]).

[3] Security whitepaper: google apps messaging and collaboration products. (Available from: http://static.googleusercontent.com/external_content/untrusted_dlcp/www.google.com/en/us/a/help/intl/en-GB/admins/pdf/ds_gsa_apps_whitepaper_0207.pdf [June 11, 2013]).

[4] Salesforce.com. (Available from: http://www.salesforce.com/ [June 11, 2013]).

[5] D. Estrin, R. Govindan, J. Heidemann, S. Kumar, Next century challenges: scalable coordination in sensor networks. In Proceedings of the 5th IEEE/ACM Annual Conference on Mobile Computing and Networks, Seattle, WA, USA, 1999:263-270.

[6] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister. System architecture directions for networked sensors. In Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems, Cambridge, MA, USA, 2000:93-104.

[7] R. H. Katz, J. M. Kahn, K.S. J. Pister, Mobile networking for smart dust. In Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, Seattle 1999, WA:271-278.

[8] J. M. Rabaey, M. J. Ammer, J.L. Silva, D. Patel, S. Roundy, PicoRadio supports ad hoc ultra low power wireless networking. IEEE Computer Magazine 2000; **33**(7):42-48.

[9] S. Kim, S. H. Son, J. A. Stankovic, S. Li, Y. Choi, SAFE: A data dissemination protocol for periodic updates in ensor networks. In Proceedings 23rd International Conf. Distributed Computing Systems Workshops (ICDCSW'03), Providence, Rhode sland, USA:19-22.

[10] F. Ye, L. Haiyun, C. Jerry, L. Songwu, L. Zhang. Sensor Networks: A two-tier data dissemination model for large-scale wireless sensor networks. In Proceedings of the Eighth Annual ACM/IEEE International Conference on Mobile Computing and Networks, Atlanta, GA, USA, 2002:148-159.

[11] H. Xuan, S. Lee S, A coordination-based data dissemination protocol for wireless sensor networks. In Proceedings of the Sensor Networks and Information Processing Conference, Brisbane, Australia, 2004:13-18.

[12] L. X. Hung, P. T. H. Truc, L. T. Vinh, A. M. Khatta. Secured WSN-integrated Cloud Computing for u-Life Care. 7th IEEE Consumer Communication and Networking Conference (CCNS 2010), USA.

[13] M. Turkanovic, B. Brumen and M. Hölbl, A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion, Ad Hoc Networks, 2014, 20, pp.96-112.

[14] S. Kalra and S. K. Sood, Advanced password based authentication scheme for wireless sensor networks, Journal of Information Security and Applications, 2015, 20, pp.37-46.

[15] B. Bllom. Space/time trade-offs in hash coding with allowable errors. Communications of the ACM 1970; **13**(7):422-426.

[16] I. Damgard, A design principle for hash function. Lecture Notes in Computer Science 1990; **435**:416-427.

[17] S. Subashini, V. Kavitha, A survey on security issues in service delivery models of cloud computing. Journal of Network and Computer Applications 2011; **34**(1):1-11.

[18] C. L. Chen, I. H. Lin, C. L. Lee and Y. F. Huang, A spare two-tier data aggregation for grid-based wireless sensor networks, Proceedings of the 6th International Wireless Communications and Mobile Computing Conference (IWCMC 2010), Caen, France, June 28- July 2, 2010, pp. 1203-1207.