A DoS-resilient enhanced two-factor user authentication scheme in wireless sensor networks

Conference Paper · February 2014

DOI: 10.1109/ICCNC.2014.6785492

CITATIONS
8

READS
47

5 authors, including:

Fei Wang
Chinese Academy of Sciences
67 PUBLICATIONS **2,355** CITATIONS
SEE PROFILE

Yujun Zhang
Chinese Academy of Sciences
77 PUBLICATIONS **1,181** CITATIONS
SEE PROFILE

Yong-Jun Xu
Chinese Academy of Sciences
243 PUBLICATIONS **4,188** CITATIONS
SEE PROFILE

Lin Wu
Chinese Academy of Sciences
27 PUBLICATIONS **498** CITATIONS
SEE PROFILE

A DoS-Resilient Enhanced Two-Factor User Authentication Scheme in Wireless Sensor Networks

Fei Wang[1,2], Yujun Zhang[1], Yongjun Xu[1], Lin Wu[1,2], Boyu Diao[1,2]
[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
Email :{ wangfei, zhmj, xyj, wulin, diaoboyu2012}@ict.ac.cn

***Abstract*—Wireless sensor networks (WSNs) are appearing to be one of the most promising pervasive applications now. In some scenarios such as commercial building surveillance or military reconnaissance, WSNs meet a lot of challenges in security, among which, user authentication is one of the most crucial. Two-factor authentication has been used in WSNs since M. L. Das's scheme in 2009, and has been attracting researchers' interest because of its robustness and flexibility, which suits resource-constrained WSNs very well. However, lots of researchers pointed out diverse security flaws in two-factor authentication scheme and came up with their improved versions. In this paper, we show that few of the existing protocols are resilient to Denial of Service (DoS) attack. And under the scenario of capturing a sensor node, some security pitfalls including gateway impersonation attack and forgery attack still exist in existing protocols. Then we propose an enhanced two-factor user authentication scheme which employs two novel techniques: lightweight pre-authentication based on Merkle hash tree and personalized secret parameters for sensor nodes. Through analysis of security and performance, we show that our proposed scheme is equipped with more security features, especially protection from DoS attack launched not only by adversaries but also by greedy users, and resilience to gateway impersonation and sensor node forgery after sensor nodes are compromised. Moreover proposed scheme maintains an acceptable performance and could adapt dynamically to DoS attacking scenarios for designated applications of WSNs.**

Keywords—Wireless sensor network; user authentication; two-factor; Merkle hash tree; DoS

I. INTRODUCTION

Nowadays, wireless sensor networks (WSNs) are widely deployed in many real-time-monitoring systems such as vehicular tracking, military surveillance, environment control, building safety monitoring, measurement of seismic activity [2, 3]. Due to pervasive distribution, easy and quick deployment, and resilience to emergent events, WSNs are gaining increasing interests from government, academia and industry. But WSNs encounters a lot challenges, among which security is the most crucial [9]. This is by reason of open communication channel, deployment in hostile areas and "attackers aren't always limited by the same constraints as the sensor devices" [6]. Furthermore, in some safety-sensitive WSN application scenarios, data are gathered through gateway nodes. These scenarios could be well targets of attackers, which highlights the significance of user authentication.

M. L. Das proposed the very first two-factor authentication protocol [4]. This protocol aimed at securing application layer security with relatively low cost compared with previous protocols [1, 9-13]. Two-factor authentication [15, 16] utilizes two-factor elements (e.g., password and smart card) which include "something you know" (user-known factor such as a password) and "something you are" (some constant or at least stubborn feature of user, such as digital certificate or biometric identifier [18]) to verify users. Since then, many researchers continually devoted themselves to improving it. But among existing protocols, security leaks in terms of one or more sensor nodes are compromised. The compromising sensor node would easily cause gateway node or sensor node impersonation attack. In addition, they are all vulnerable to denial of service (DoS) attack performed by adversaries or greedy users in login phase. Details about security leaking will be revealed later.

In this paper, we propose a DoS resilient enhanced two-factor authentication scheme in WSNs. It inherits benign attributes of two-factor authentication and attains more secure features through techniques of lightweight pre-authentication based on Merkle hash tree and personalized secret parameters for sensor nodes. It is able to resist gateway node or sensor node impersonation after sensor nodes are compromised. To the best of our knowledge, the proposed scheme is the first that is able to prevent DoS attack from both unauthorized and greedy users.

Rest of paper is arranged as follows: Section-II reviews related work. Section-III presents the network model and adversary model. Section-IV introduces related notations and offers cryptanalysis of Vaidya et al.'s scheme. Section-V presents and discusses the proposed scheme and Section-VI analyzes it. Section-VII concludes the paper.

II. LITERATURE REVIEW

There existed diverse authentication ideas before two factor authentication. Benenson et al. proposed a scheme based on Public Key Center (PKC) and Elliptic Curve Cryptography (ECC) in [11]. Later Watro et al. presented a public key based scheme called TinyPK [10], which is based on RSA. However both schemes have some security pitfalls including sensor node impersonation and DoS. Additionally, computation cost of their schemes is high and cannot suit WSNs very well. Wong et al. [1] proposed a dynamic user authentication scheme for WSNs which utilizes only one-way hash function and exclusive-OR operations. This reduces computation overhead. However it could not resist replay, forgery, stolen-verifier and password guessing. Tseng et al. [12] revealed security flaws in Wong et al.'s scheme and provided an improved protocol

This paper is partially sponsored by National Basic Research Program of China (973 Program) (No. 2012CB315804), Important National Science & Technology Specific Projects (No. 2014ZX03006-003), National High-tech R&D Program of China (863 Program) (No. 2011AA010703), National Natural Science Foundation of China (NSFC) (No. 61173132, No. 61100177, No. 61173133), and Special IOT Program of China's National Development and Reform Commission. As for all content of this paper, the authors alone take the responsibilities.

which could protect from replay and forgery. Plus, the protocol allows users to update their passwords. Afterwards, L. C. Ko [13] picked out some weakness of Tseng et al.'s work, caused by the lack of mutual authentication between sensor nodes and gateway nodes or users. Vaidya et al. [20] cryptanalyzed the authentication schemes of Wong et al. and Tseng et al. and proposed their improvements in resistance to replay attack, forgery attack and man-in-the-middle attack, and mutual authentication between users and gateway node.

The aforementioned schemes have problems in flexibility and efficiency. What's more, all of them have obvious security loopholes. Das's scheme [4] started the utilization of two factor authentication in WSNs. It employs a password held by user, a smart card and a secret value shared between gateway node and designated sensor nodes. It is implemented through only one-way hash functions and exclusive-OR operations and achieves high efficiency. This protocol could resist attacks like many logged in users with the same login-id, replay, stolen-verifier, off-line password guessing, yet it could not resist insider offline guessing, gateway impersonation, sensor node forgery, DoS, and node compromising. After Das, researchers continue improving the two factor authentication in WSNs. Nyang and Lee proposed an enhanced two-factor user authentication protocol for WSNs [5], which drives off security flaws like insider offline guessing, sensor node compromising from Das's scheme. Khan and Alghathbar [20] identified several other flaws of Das's scheme including privileged-insider attack, gateway bypass attack, no provision for updating passwords of users, absence of mutual authentication between gateway node and sensor nodes. Then they proposed an improved two-factor user authentication in WSNs which is able to resist to gateway-node bypassing attack and allows users to update passwords. More recently, Vaidya et al. [14] pointed out that even using tamper-proof technology, invasive attacks or side channel attacks [22] would still cause the secret value leaking and lead to potential threats. As a result, they found out some flaws from [4] and [20] including stolen smart card attack, gateway impersonation with node captured and forgery attack with node captured. Then they subdued the harmful threats and announced their improved two-factor user authentication in WSNs. Yet, security leaks in terms of node captured during the interval of two password changes, which will be discussed later in Section IV. Not long ago, Kumar et al. [17, 21] also inherited the Das-scheme and offered their solution. Resilience to many attacks is claimed, such as many-logged in users attack, replay attack, impersonation attack, stolen-verifier attack, password guessing attack, node-compromise attack, man-in-the-middle attack and denial of service attack. While in fact, their scheme could not resist insider's attack, stolen smart card attack and sensor node compromised attack, same as in Vaidya et al.'s scheme. What's more, in their scheme the resilience to DoS is limited to password-changing phase.

III. System Model and Adversary Model

A. System Model

In this paper, we focus on certain WSN application scenarios such as surveillance for a large commercial building or military reconnaissance, in which data are sensitive and WSNs might well be targets of adversarial attacks. As for network model, we consider a large data gathering network which is composed of several independent WSNs. Each WSN comprises a large number of cheap sensor nodes with limited memory space and computation power (e.g., widely used TELOSB sensor node with an 8MHz microcontroller and with 10kB RAM) distributed over a sensing field. We suppose there exists a gateway node for each WSN, which is responsible for verifying users and switching queries and sensor data. The gateway node here is slightly more powerful than sensor nodes in storage and computational capability (e.g., Imote2 sensor node with a 13~416MHz microcontroller and 256kB SRAM). It also has a human-machine interface for a user to plug his smart card and input password. Each user could separately read the data from each WSN through a gateway node after passing the authentication. In some scenarios, wireless sensors, switching nodes, and users of the sensor data are integrated into a single large wireless communications network (e.g., a tactical military network), but it is not included it in this paper.

B. Adversary Model

We assume the adversary has powerful computational capability and communication capability. It could control the whole communication channel, monitor the on-the-fly data, tamper messages and even replace the original data.

What's more, adversaries are able to steal somebody's smart card and compromise one or more sensor nodes (at most 5% of all). After compromising and tampering a smart card, an adversary would try to disguise himself into a legitimate user to use the network resources or attack the network. User fully compromised with not only his smart card but also his password, is not considered in this paper. We assume compromising one or more sensor nodes and acquiring information from them is possible. And with parameters extracted from them, an adversary might try to disguise himself into forge bogus sensor information to mislead the users of WSNs. Security of data transfer between users and sensor nodes is not concerned in this work.

An adversary might try to bypass the gateway node and interact directly with sensor nodes to acquire data from WSNs. Nonetheless, gateway node is infeasible for any adversary to compromise due to our assumption that gateway node is geographically fixed and hard to compromise (e.g., human-machine terminal built into the commercial building structure).

We assume adversaries could launch DoS attack by sending large quantities of invalid login messages to gateway node and disable the normal service. Moreover, legitimate users could also be the launcher of DoS attack by abusing gateway node of WSNs and exhaust the gateway resources. In actual WSNs, straightforward denial-of-service attack could be accomplished by jamming, disrupting information transfer among users, sensor nodes, and the gateway. Countering such attacks is outside the scope of this paper.

Adversaries mentioned in this paper are literally much stronger than ones in existing schemes.

IV. CRYPTANALYSIS OF VAIDYA ET AL.'S SCHEME

In this section, Vaidya et al.'s scheme will be cryptanalyzed. Notations to be used in this paper are shown in TABLE I.

TABLE I NOTATIONS USED IN PROPOSED SCHEME

Symbol	Description
UD	User of WSNs
SN	Sensor node
GWN	Gateway node
ID_x	Identity, i means user, s means smart card
DID_i	Dynamic user identity
pw_i	Password selected by user
S_n	Sensor node identity
K_g	Secret key known to GWN only
x_s	Secret value generated by the GWN and stored securely in designated SN
$h(\cdot)$	One-way hash function
$\|$	Bit-wise concatenation operator
$\oplus$	Exclusive-OR operation
=?	Verification operation
T_x	Current timestamp: x=1, 2, 3…
ΔT	Expected time interval for the transmission delay
$Cert_i$	Certificate distributed to U_i
key_i^j	The j-th key of the i-th certificate($Cert_i$)

Vaidya's scheme [14] improves the security on smart card by adding a verification of password. Thus it could provide new security features as secure change of password and resilience to attack with stolen smart card. What's more, it is also the first two-factor authentication scheme to realize mutual authentication between a gateway node and sensor nodes. The scheme is claimed to resist gateway impersonation and sensor node forgery even with sensor node captured and secret value leaking due to secure update of password. Yet we cannot guarantee that users would always update their passwords after every query. Accordingly, any adversary is free to intrude the security by utilizing the time interval between any two sequential passwords. What's more, Denial-of-service is not taken into consideration in Vaidya et al.'s scheme. Details of aforementioned attacks and cryptanalysis are as follows:

1) Gateway impersonation attack with node captured: with $\{S_n, x_s\}$ extracted from an compromised sensor node and a valid login message $\{DID_i, \varepsilon_i, T\}$, an adversary could attack WSNs between any two sequent changes of user's passwords:

- Compute $\sigma_{ei} = h(DID_i \| S_{en} \| x_s \| T_{e1})$ (T_{e1} used here is some current timestamp much larger than T, S_{en} here is any arbitrary sensor node identity the adversary wants to acquire sensitive information from)
- Send $\{DID_i, \sigma_{ei}, T_{e1}\}$ to SN with id S_{en}

Upon receiving the message from GWN, SN is supposed to carry out the following operations:

- Verify $T_{e2} - T_{e1} \le \Delta T$
- Compute $\sigma_{ei}^* = h(DID_i \| S_{en} \| x_s \| T_{e1})$
- Verify $\sigma_{ei}^* = ? \sigma_{ei}$ which absolutely returns equality
- Compute $\mu_{ei} = \sigma_{ei}^* \oplus x_s$
- Compute $\omega_{ei} = h(\mu_i \| x_s \| T_{e2})$
- Send $\{\omega_{ei}, T_{e2}\}$ to adversary-impersonating GWN

At last, adversary pretends to receive message from SN, and needs only to send command accept login to SN which would believe that the command is from a valid GWN and starts the delivery of sensor data. Adversary then could get any sensitive data it wants from any SN.

2) Forgery attack with node captured: Vaidya claimed his scheme's resilience to forgery attack along with node captured, while in fact there lies two different ways to attack. The preparations include both compromising-attaining $\{S_n, x_s\}$ and eavesdroping-attaining $\{DID_i, \varepsilon_i, T\}$.

The first kind is performed as followings:

- Listen and get a message $\{DID_i, \sigma_i, T_1\}$ from GWN.
- Compute $\mu_{ei} = \sigma_i \oplus x_s$ and $\omega_{ei} = h(\mu_{ei} \| x_s \| T_{e2})$
- Send $\{\omega_{ei}, T_{e2}\}$ to GWN

GWN receives the message from adversary and pass the verification $\omega_{ei}^* = ? \omega_{ei}$. This kind of forgery is limited because an adversary can only pretend as SN with identity S_n rather than with any S_{en}. As a result, such one-replace-one forgery could not invoke significantly terrible disturbance in WSNs.

The other way which is complicated but more powerful is executed like follows:

- Compute $\sigma_{ei} = h(DID_i \| S_{en} \| x_s \| T_{e1})$ (S_{en} is arbitrary sensor node identity the adversary wants to impersonates as)
- Compute $\mu_{ei} = \sigma_{ei} \oplus x_s$
- Wait for whenever user wants to query data from sensor node from SN with identity S_{en} and compute $\omega_{ei} = h(\mu_{ei} \| x_s \| T_{e2})$, then send $\{\omega_{ei}, T_{e2}\}$ to GWN

GWN would also let the verification pass just as the first above forgery method, but this is more harmful to WSNs because it is able to impersonate as any authorized SN with any desirable sensor node identity.

3) Denial of service attack: Kumar et al. [17] claimed DoS resilience on stolen smart card of their schema in the sense that verifying ID_i and pw_i first in password changing phase would prevent the unauthorized user from changing new password. But it is not enough. Vaidya et al.'s scheme also realized the phase for users to update passwords and gained the benefits. However, there still exists possibilites of DoS in

other phases. Vaidya et al.'s scheme cannot resist the two kinds of DoS attack discussed in Section-III.

Considering Vaidya et al.'s scheme's behaviors in terms of login and authentication phase, it could resist adversary-launch DoS started from the beginning step of login phase but it might not protect the login request step of the stage. The former type of DoS is because smart card would validate ID_i and pw_i first. No matter how many requests adversary performs to smart card, it could never influence *GWN*. The latter is due to the reason that if adversary jumps over the step of smart card verification and directly sends a large amount of false login request messages, *GWN* would be busy verifying them and has no resources to provide service to legitimate users. What's more, greedy-user-launch DoS attack will be nearly two times much influential than Das's scheme because the computation overhead is about two times higher.

V. PROPOSED SCHEME

Our proposed two-factor authentication scheme employs two enhancing techniques: 1) Lightweight pre-authentication using Merkle hash tree. A Merkle hash tree is constructed by gateway node in initialization phase and used in login/authentication phase to perform pre-authentication consuming only one hash operation, which could not only prevent adversary performing DoS attack but also set up strict access control for malicious registered users. 2) Personalized secret parameters for sensor nodes. Traditional shared value between gateway node and all sensor nodes makes previous schemes vulnerable to a variety of attacks caused by sensor node captured. The proposed scheme individualizes the secret value for each sensor node, which disables such secret leakage while maintaining the features of two factor authentication.

A. Initialization Phase

A lightweight pre-authentication is introduced at gateway based on Merkle hash tree which is firstly used in WSNs as an access control technique by Shen et al [23]. When the gateway node and sensor nodes are deployed, *GWN* constructs a Merkle hash tree as an access token chain generator and stores it. The details of constructing such a tree could be found in [23]. Creating a Merkle hash tree of 2^n certificates needs $(2^n - 1)$ hash operations. Initially, every user is assigned with an access token chain and every key in the chain could be verified. Thus, such a chain could be used as a limit-count commitment of valid accesses. What's more such a structure is nearly two times efficient in terms of storage compared with storing one separate key chain for every user. After all the keys from a chain is verified, root value is exposed and the certificate lose effectiveness In Fig. 1., a tree supporting 8 certificates is created and $Cert_1 = \{C_1, K_2, K_{3,4}, K_{5,8}, K_{1,8}\}$ is distributed to the first user, which could be authenticated 4 times as shown in (1).

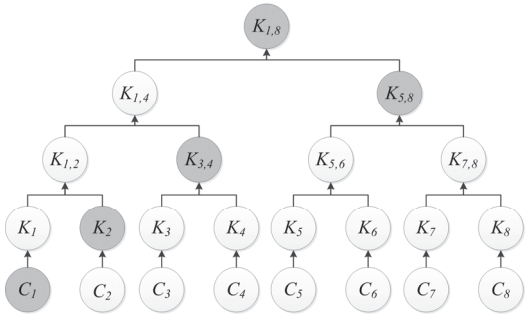

Fig.1. Example of a Merkle hash tree based commitment distribution

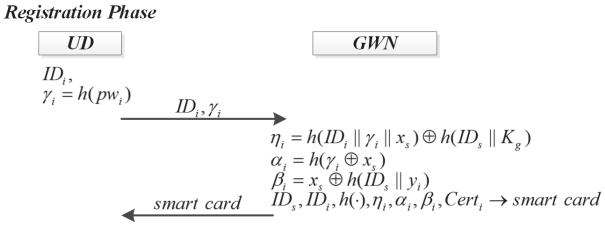

Fig. 2. Registration phase of proposed scheme

$$K_{1,8} = h(h(h(h(C_1) \| K_2) \| K_{3,4}) \| K_{5,8}) \qquad (1)$$

Another technique is that gateway node would calculate a personalized x_s for each sensor node. A typical example is like $h(S_n \| x_s)$. Then the individualized security parameter is written into storage of mapping node as preparation of further realization of mutual authentication between *GWN* and *SN*.

B. Registration Phase

A user U_i would compute $\gamma_i = h(pw_i)$ and send $\{ID_i, \gamma_i\}$ as request for registration to *GWN*. Having received the request, *GWN* would compute $\eta_i = h(ID_i \| \gamma_i \| x_s) \oplus h(ID_s \| K_g)$, then $\alpha_i = h(\gamma_i \oplus x_s)$ and $\beta_i = x_s \oplus h(ID_s \| \gamma_i)$. He then picks a $Cert_i$ for U_i and stores the mapping relation. Having writen $ID_s, ID_i, h(\cdot), \eta_i, \alpha_i, \beta_i, Cert_i$ into a smart card, *GWN* would send the smart card to user U_i. The process is shown in Fig. 2.

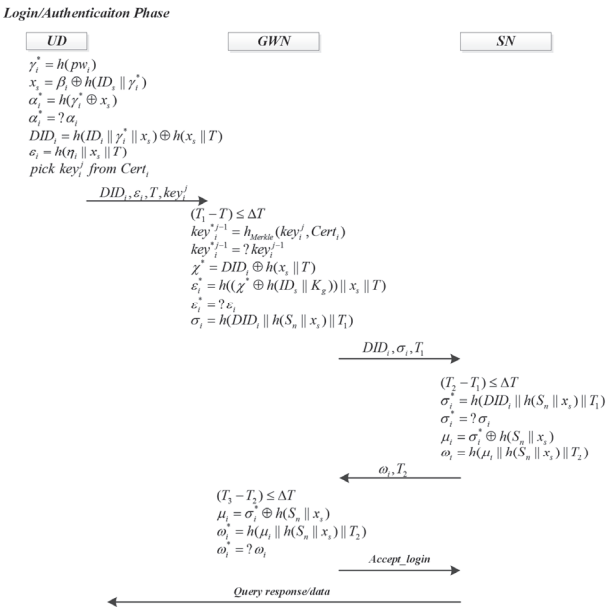

Fig. 3. Login/Authentication phase of proposed scheme

C. Login/Authentication Phase

When *UD* intends to query or access sensor data from the WSNs, the login/authentication phase begins. *UD* firstly plugs his smart card into the physical terminal. Then he inputs his ID_i and pw_i. Through cooperation of the terminal and the smart card, *UD* will firstly verify the password by $\alpha_i^* = ?\alpha_i$. Then *UD* gets its dynamic identity DID_i and security parameter ε_i. Along with the picked live key key_i^j from $Cert_i$, *UD* would form a login request and send it to *GWN*.

After receiving the message of login, *GWN* would execute the pre-authentication: it firstly verifies the key_i^j by computing $key^{*j-1}_i = h_{Merkle}(key_i^j, Cert_i)$ which means the next key of key_i^j in $Cert_i$, namely the father node of key_i^j in tree. If $key^{*j-1}_i = ?key_i^{j-1}$ returns false, the login is rejected. Otherwise, key_i^j is killed. After this pre-authentication, *GWN* continues the authentication as is shown in Fig. 3. If core authentication is passed, *GWN* would send $\{DID_i, \sigma_i, T_1\}$ to *SN* and invoke the authentication on sensor nodes.

Upon receiving the message, *SN* authenticates the messages through computing secret value $\sigma_i^* = h(DID_i \| h(S_n \| x_s) \| T_1)$ and verifying $\sigma_i^* = ?\sigma_i$. If false, it aborts the process, otherwise it prepares revise authenticating messages $\{\omega_i, T_2\}$ and send it back to *GWN* to invoke the final part of mutual authentication.

GWN then authenticates the sensor node's identity through computing $\mu_i = \sigma_i^* \oplus h(S_n \| x_s)$, $\omega_i^* = h(\mu_i \| h(S_n \| x_s) \| T_2)$ and then verifying $\omega_i^* = ?\omega_i$. If returns false, *GWN* sends a termination message, otherwise send *SN* an acceptance message.

Having received an acceptance message from *GWN*, *SN* would respond to *UD*'s query request with designated data. This phase is shown in Fig. 3.

D. Password Change Phase

In our protocol, registered users could change or update passwords without interacting with gateway node.

The change of password needs a valid user to input his current password pw_i and new password pw_i' firstly. Then pw_i will be confirmed by verification of $\alpha_i^* = ?\alpha_i$ as in login/authentication phase. If pw_i is proved invalid, password change request is aborted. If pw_i is valid, smart card performs:

- Compute $\gamma_i' = h(pw_i')$
- Compute $\eta_i' = \eta_i \oplus h(ID_i \| \gamma_i \| x_s) \oplus h(ID_i \| \gamma_i' \| x_s)$
- Compute $\alpha_i' = h(\gamma_i' \oplus x_s)$, $\beta_i' = x_s \oplus h(ID_s \| \gamma_i')$
- Replace $\eta_i, \alpha_i, \beta_i$ with $\eta_i', \alpha_i', \beta_i'$

Therefore, the change process of password is completed and smartcard is updated, which could be used in further process of login/authentication phase.

VI. Analysis of the Protocol

In this section security and performance analysis of our proposed two-factor authentication protocol will be presented.

A. Security Analysis

1) Brute force attack: it is preventable due to that neither *GWN* nor smart card contains password/verifier. What's more, even direct hash value of password is not stored in the above two entities.

2) Gateway node bypass attack: our scheme is resilient to *GWN* bypass attack because x_s is not directly written to smart card, which makes it difficult to calculate a dynamic identity DID_i. Besides, responses to *UD*'s query needs an acceptance message from gateway node.

3) Insider's attack: user only sends hashed password in registration phase, which makes it impossbile to attack proposed protocol as an insider.

4) Stolen smart card attack: suppose that an adversary steals a smart card, he could extract secret parameters $ID_s, ID_i, h(\cdot), \eta_i, \alpha_i, \beta_i, Cert_i$ from it. To impersonate a

TABLE II Security Features Comparison

Security Features/Main Existing Schemes	Wong et al.'s [1]	Das's [4]	Nyang et al.'s [5]	He et al.'s [7]	Khan et al.'s [8]	Vaidya et al.'s [14]	Kumar et al.'s[17]	Proposed
Resist to many logged-in with same id attack	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Resist to brute force attack	No	Partial	Partial	Yes	Yes	Yes	Partial	Yes
Resist to gateway bypass attack	No	No	No	No	Yes	Yes	Yes	Yes
Resist to privileged insider's attack	No	No	No	Yes	Yes	Yes	No	Yes
Zero-knowledge password protocol	No	No	No	Yes	Yes	Yes	No	Yes
Securely change/update of password	No	No	No	Yes	Yes	Yes	Yes	Yes
Mutual authentication between *GWN* and *SN*	No	No	Yes	No	Partial	Yes	Yes	Yes
Resist to *GWN* impersonation with node captured	No	No	Yes	No	Yes	Partial	Partial	Yes
Resist to forgery attack with node captured	N/A	N/A	No	N/A	No	No	No	Partial
Resist to stolen smart card attack	No	No	No	No	No	Yes	No	Yes
Resist to DoS attack with stolen smart card	N/A	N/A	N/A	No	No	Yes	Yes	Yes
Resist to adversary DoS attack in login phase	N/A	N/A	N/A	N/A	N/A	No	No	Yes
Resist to greedy-user DoS attack in login phase	No	No	No	No	No	No	No	Yes

legitimate user, it is essential to generate $\{DID_i, \varepsilon_i, T, key_i^j\}$ as a valid login request message. However, without pw_i which is only kept by users, adversary can never obtain x_s from β_i and ID_s. Thus, neither DID_i nor ε_i can be computed which makes it infeasible to form a valid login request.

5) Zero knowledge password protocol: the proposed scheme is zero knowledge password protocol, due to that any legitimate user could authentiate himself to *GWN* while keeping his password secret and not exposed.

6) Mutual authentication between GWN and SN: GWN and *SN* could mutually authenticate between each other in our scheme, which is robust and flexible enough to resist sensor node impersonation.

7) Gateway impersonation attack with node captured: The key to realize gateway impersonation is to calculate a valid $\sigma_{ei} = h(DID_i \| S_{en} \| x_s \| T_{e1})$ which needs extraction of $\{S_n, x_s\}$ from a compromised sensor node.

In the proposed scheme, x_s is personalized as $h(S_n \| x_s)$ and stored in sensor nodes. Due to the preimage resistant feature of one-way hash function, it is difficult to acquire x_s from extracted $h(S_n \| x_s)$. Correspondingly, gateway impersonation attack analyzed in Section-IV would never be successful when performed on sensor nodes with identity other than S_n. What's more, timely change of password makes it even harder to get $h(ID_i \| \gamma_i^* \| x_s)$ or a valid dynamic identiy because γ_i^* along with pw_i. As discussed after a node has been compromised, an adversary cannot impersonate the gateway even in interval of two updates of passwords.

8) Forgery attack with node captured: suppose adversary has extracted $\{S_n, h(S_n \| x_s)\}$ from a captured sensor node. The simple but limited type of forgery attack pointed out in Section-IV is not concerned in this paper because it affects data from only one sensor node. As for the second type aforementioned forgery which is much more harmful, it would impossibly work out for adversary. The reason is basically the same as scenery facing gateway impersonation that without the knowing of x_s, it's hard to get a valid DID_i, nor a valid σ_i or μ_i. Consequently, an adversary could not counterfeit $\{\omega_i, T_2\}$ for a designated fake sensor node identity S_{en}. In a nutshell, our scheme is partially resilient to forgery attack after one or more sensor nodes are captured.

9) Denial of service attack: denial of service attack might happen in two forms as talked about in Section-III. The first is performed by an adversary. However in our proposed protocol, key from Merkle hash tree is verified firstly, which uses only one hash function to protect the core calculation process of login/verification. The second form is performed by a greedy legitimate user through querying WSNs a large quantity of times in a short time interval. Yet in our scheme, access token chain consists of a number of keys which could be used to put a limit to a valid user's accesses. Each query would kill a key synchronously at smart card and *GWN*. Running out of keys, the user has to update his smart card in front of gateway node, which would clearly imposes his maliciousness.

There are doubts whether use of Merkle hash tree causes high price in space and time, which is absolutely rational. We tested that the time cost for one hash function (here SHA-1 is used as our hash function) on TELOSB node is about 0.540ms and can infer that creating a Merkle tree of 256 certificates on Imote2 node costs less than 276ms. Considering the storage overhead, it consumed about 22kB. Obviously, Imote2 node could afford the time and space cost. What's more, creation is offline which makes no effects on the service progress. As for lookup, only $O(1)$ is needed to discriminate two certificates and $O(\log N)$ to locate one access token in Merkle hash tree, making it stand out to reach a better tradeoff.

Shown in TABLE II, our protocol has realized nearly all of aforementioned security features and shows out to be the most robust compared with most of the existing two-factor solutions. Yet, our scheme cannot resist forgery attack completely, and it can provides neither secure channel for query responses nor mutual authentication between users and sensor nodes.

TABLE III. COMPUTATIONAL COST COMPARISON

Scheme/Phase	Registration			Login/Authentication		
	UD	***GWN***	***SN***	***UD***	***GWN***	***SN***
Wong et al.'s [1]		3 t_h			1 t_h 2 t_{XOR}	3 t_h 2 t_{XOR}
Das's [4]		3 t_h 1 t_{XOR}		3 t_h 1 t_{XOR}	4 t_h 2 t_{XOR}	1 t_h
Nyang et al.'s [5]		3 t_h 1 t_{XOR}		6 t_h 1 t_{XOR} (1 t_{DEC})	8 t_h 2 t_{XOR} (1 t_{ENC})	4t_h (1 t_{ENC}) (1 t_{DEC})
He et al.'s [7]	1 t_h 1 t_{XOR}	5 t_h 2 t_{XOR}		2 t_h 1 t_{XOR}	4 t_h 2 t_{XOR}	1 t_h
Khan et al.'s [8]	1 t_h	2 t_h 1 t_{XOR}		4 t_h 1 t_{XOR}	5 t_h 2 t_{XOR}	2 t_h
Vaidya et al.'s [14]	1 t_h	4 t_h 3 t_{XOR}		6 t_h 2 t_{XOR}	5 t_h 3 t_{XOR}	2 t_h 1 t_{XOR}
Kumar et al.'s[17]		3 t_h		3 t_h 1 t_{XOR}	4 t_h 1 t_{XOR}	2 t_h
Proposed	1 t_h	4 t_h 3 t_{XOR}		6 t_h 2 t_{XOR}	7 t_h 3 t_{XOR}	2 t_h 1 t_{XOR}

B. Performance Analysis

Metrics used to evaluate performance is computational cost of different security entities in each phases. In TABLE III, we use t_h and t_{XOR} to represent time cost of accomplishing one-way hash function and an exclusive-OR operation. As above chapter, t_h is 0.540ms and t_{XOR} is 0.111ms on TELOSB node. As in Nyang et al.'s scheme, t_{ENC} and t_{DEC} are time consumed by encryption and decryption of AES respectably, which are much higher than t_h and t_{XOR}.

From TABLE III, Nyang et al.'s scheme achieves mutual authentication and secure channel between users and sensor nodes, but it is the most inefficient. The total computational cost of proposed scheme is nearly the same with that of Khan et al.'s and Vaidya et al.'s, and is a little higher than others.

We also compared the total number of login messages that the gateway node could authenticate in 1 second along with the growth of percentage of invalid login messages. As is shown in Fig. 4, although the proposed scheme has a little

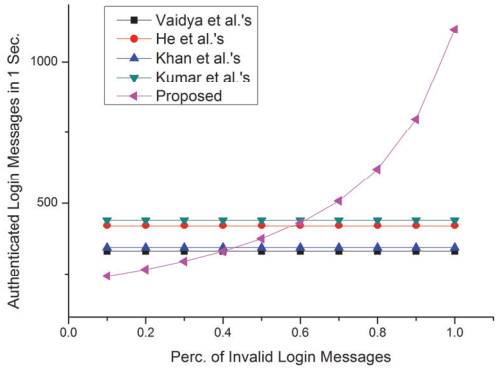

Fig. 4. Comparison of authenticated login messages in 1 sec.

lower efficiency than the others when the percentage of invalid login messages is smaller than 40%, it could achieve high efficiency even with 90% to 100% invalid login messages. For example, Vaidya et al.'s scheme could only verify about 330 login requests every second ignoring the ratio of false ones. The proposed schema could verify about 1120 requests every second, which is nearly 3.3 times efficient than Vaidya et al.'s scheme. Besides, none of existing schemes except the proposed one is able to adapt to DoS-attacking scenario dynamically which has practical meanings in WSNs.

VII. CONCLUSION

In this paper, we have compared most of related two-factor user authentication schemes for WSNs and analyzed Vaidya et al.'s. Through analysis, it could be observed that nearly all of the schemes are not resilient to DoS attack, and under the circumstance of sensor node captured, some security pitfalls including gateway impersonation and forgery still exist in those schemes. Aiming at overcoming these security flaws, we proposed a DoS-resilient enhanced two-factor authentication scheme which features access control based strategic Merkle hash tree and personalized secret parameter shared between *GWN* and *SN*. Afterwards, through security and performance analysis, our proposed schemes are proved capable of resisting to nearly all of mentioned attacks with an acceptable efficiency, moreover it could achieve high efficiency even with 90% to 100% invalid login messages and adapt dynamically to DoS attacking scenarios. It is also the first two-factor authentication scheme which solves the access control on greedy users of WSNs who would perform DoS.

REFERENCES

[1] K. H. M. Wong, Z. Yuan, C. Jiannong, and W. Shengwei, "A dynamic user authentication scheme for wireless sensor networks," in Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on, 2006, p. 8 pp.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer Networks, vol. 38, pp. 393-422, 2002.

[3] C. Chee-Yee and S. P. Kumar, "Sensor networks: evolution, opportunities, and challenges," Proceedings of the IEEE, vol. 91, pp. 1247-1256, 2003.

[4] M. L. Das, "Two-factor user authentication in wireless sensor networks," Wireless Communications, IEEE Transactions on, vol. 8, pp. 1086-1090, 2009.

[5] D. Nyang and M.-K. Lee, "Improvement of Das's Two-Factor Authentication Protocol in Wireless Sensor Networks.," Cryptology ePrint Archive, 2009.

[6] D. R. Raymond and S. F. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses," Pervasive Computing, IEEE, vol. 7, pp. 74-81, 2008.

[7] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu, "An Enhanced Two-factor User Authentication Scheme in Wireless Sensor Networks," Ad Hoc & Sensor Wireless Networks, vol. 10, pp. 361-371, 2010.

[8] M. K. Khan and K. Alghathbar, "Cryptanalysis and Security Improvements of 'Two-Factor User Authentication in Wireless Sensor Networks'," Sensors, vol. 10, pp. 2450-2459, 2010.

[9] Z. Benenson, F. Garther, and D. Kesdogan, "User authentication in sensor networks," in Informatic 2004, Workshop on Sensor Networks, 2004.

[10] R. Watro, D. Kong, S.-f. Cuti, C. Gardiner, C. Lynn, and P. Kruus, "TinyPK: securing sensor networks with public key technology," presented at the Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks, Washington DC, USA, 2004.

[11] Z. Benenson, N. Gedicke, and O. Raivio, "Realizing robust user authentication in sensor networks.," in REALWSN 2005, 2005.

[12] T. Huei-Ru, J. Rong-Hong, and Y. Wuu, "An Improved Dynamic User Authentication Scheme for Wireless Sensor Networks," in Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE, 2007, pp. 986-990.

[13] K. Lee-Chun, "A novel dynamic user authentication scheme for wireless sensor networks," in Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on, 2008, pp. 608-612.

[14] B. Vaidya, D. Makrakis, and H. T. Mouftah, "Improved two-factor user authentication in wireless sensor networks," in Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on, 2010, pp. 600-606.

[15] G. Yang, D. S. Wong, H. Wang, and X. Deng, "Two-factor mutual authentication based on smart cards and passwords," Journal of Computer and System Sciences, vol. 74, pp. 1160-1172, 2008.

[16] D. Coffin, "Two-Factor Authentication Expert Oracle and Java Security," ed: Apress, 2011, pp. 177-208.

[17] P. Kumar, M. Sain, and L. Hoon Jae, "An efficient two-factor user authentication framework for wireless sensor networks," Proceedings of the 2011 13th International Conference on Advanced Communication Technology (ICACT). Smart Service Innovation through Mobile Interactivity, pp. 574-578578, 2011.

[18] A. T. B. Jin, D. N. C. Ling, and A. Goh, "Biohashing: two factor authentication featuring fingerprint data and tokenised random number," Pattern Recognition, vol. 37, pp. 2245-2255, 2004.

[19] B. Vaidya, J. J. Rodrigues, and J. H. Park, "User authentication schemes with pseudonymity for ubiquitous sensor network in NGN," International Journal of Communication Systems, vol. 23, pp. 1201-1222, 2010.

[20] M. K. Khan and K. Alghathbar, "Cryptanalysis and Security Improvements of 'Two-Factor User Authentication in Wireless Sensor Networks'," Sensors, vol. 10, pp. 2450-2459, Mar 2010.

[21] P. Kumar and L. Hoon-Jae, "Cryptanalysis on two user authentication protocols using smart card for wireless sensor networks," 2011 Wireless Advanced (WiAd 2011), pp. 241-245245, 2011.

[22] K. Markantonakis, M. Tunstall, G. Hancke, I. Askoxylakis, and K. Mayes, "Attacking smart card systems: Theory and practice," Information Security Technical Report, vol. 14, pp. 46-56, 2009.

[23] Y. Shen, J. Ma, and Q. Pei, "An Access Control Scheme in Wireless Sensor Networks," in Network and Parallel Computing Workshops, 2007. NPC Workshops. IFIP International Conference on, 2007, pp. 362-367.