Understanding Node Capture Attacks in User Authentication Schemes for Wireless Sensor Networks

Chenyu Wang, Ding Wang, Yi Tu, Guoai Xu, and Huaxiong Wang

Abstract—Despite decades of intensive research, it is still challenging to design a practical multi-factor user authentication scheme for wireless sensor networks (WSNs). This is because protocol designers are confronted with a long-standing "security versus efficiency" dilemma: sensor nodes are lightweight devices with limited storage and computation capabilities, while the security requirements are demanding as WSNs are generally deployed for sensitive applications. Hundreds of proposals have been proposed, yet most of them have been found to be problematic, and the same mistakes are repeated again and again. Two of the most common security failures are regarding smart card loss attacks and node capture attacks. The former has been extensively investigated in the literature, while little attention has been given to understanding the node capture attacks. To alleviate this undesirable situation, this article takes a substantial step towards systematically exploring node capture attacks against multi-factor user authentication schemes for WSNs. We first investigate the various causes and consequences of node capture attacks, and classify them into ten different types in terms of the attack targets, adversary's capabilities and vulnerabilities exploited. Then, we elaborate on each type of attack through examining 11 typical vulnerable protocols, and suggest corresponding countermeasures. Finally, we conduct a large-scale comparative measurement of 61 representative user authentication schemes for WSNs under our extended evaluation criteria. We believe that such a systematic understanding of node capture attacks would help design secure user authentication schemes for WSNs.

Index Terms—User authentication, node capture attacks, wireless sensor networks

1 Introduction

The past ten years have witnessed the prosperity and development of wireless sensor networks. As the elementary infrastructure of Internet of Things, WSNs are widely used in smart homes [1], public safety [2], personal health [3] and intelligent transportation systems [4]. A WSN is an ad-hoc network consisting of a large number of sensor nodes which are connected by wireless communication. These sensor nodes can collaboratively monitor information from network coverage area [5], and typically external parties are allowed to access the real-time data in sensor nodes to acquire the status of the monitoring entity [6], [7]. As such, it is critical that the sensitive data are not accessed by malicious adversaries. Therefore, a well-designed user authentication method is necessary.

- *C. Wang and G. Xu are with the School of Cyber Security, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the National Engineering Laboratory of Mobile Network Security, Beijing 100876, China. E-mail: {wangchenyu, xga}@bupt.edu.cn.*
- *D. Wang is with College of Cyber Science, Nankai University, Tianjin 300350, China, with the State Key Laboratory of Cryptology, Beijing 100878, China, and also with Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin 300350, China. E-mail: wangding@nankai.edu.cn.*
- *Y. Tu and H. Wang are with the School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371. E-mail: tuyi0002@e.ntu.edu.sg, hxwang@ntu.edu.sg.*

Manuscript received 12 Sept. 2019; revised 18 Jan. 2020; accepted 11 Feb. 2020. Date of publication 17 Feb. 2020; date of current version 17 Jan. 2022.
(Corresponding author: Ding Wang.)
Digital Object Identifier no. 10.1109/TDSC.2020.2974220

Generally, there are three factors used for authenticating a person: something only she knows, such as a password [8]; something she has, such as a smart card; something she is, such as a biometric trait [9]. Due to their simplicity and convenience, password-based authentication protocols get quite popular [10]. Smart card and biometric factors are usually added to password-based protocols as a way for increasing security [1], [11]. A protocol which combines at least two factors is called a multi-factor user authentication protocol. It is typically used for security-crucial systems, such as wireless sensor networks as shown in Fig. 1. In this authentication models, three participants are included: 1) A set of users U, who may want to access the real-time data from a sensor node, and 2) A large number of distributed sensor nodes SN, which are deployed to detect, monitor and collect data, and may help to process the data; 3) The gateway GWN, who acts as a controller, provides a registration service, and is a communication bridge between users and sensor nodes. Though our results can be also applied to multi-gateway environments, this paper primarily focuses on the single-gateway architecture as shown in Fig. 1. Besides, unless otherwise specified, the figures, tables and various conclusions in this article are for multi-factor user authentication in WSNs.

The request for a user authentication protocol that ensures the security of communication and avoids eavesdropping by adversaries, has resulted in a large number of proposals. However, designing a multi-factor user authentication scheme for WSNs is full of challenges due to the fact that the protocol designer is confronted with a powerful adversary, resource-

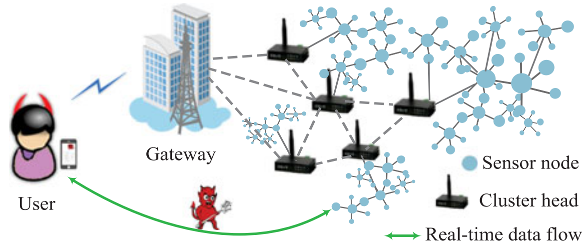


Fig. 1. User Authentication in wireless sensor networks (WSNs).

constrained hardwares and demanding security requirements. To reveal the difficulties in designing a multi-factor user authentication scheme for WSNs, we revisit dozens of typical schemes and identify their security flaws. For a concrete grasp, we show the result in Fig. 2, illustrating the development history of multi-factor user authentication protocols for WSNs. It can be seen that most proposals have been found insecure or unable to provide certain important security attributes. Particularly, most schemes are unable to resist offline dictionary attacks or node capture attacks. Among these examined schemes, only two are secure against node capture attacks.

To repaire offline dictionary attacks, much effort has been made. Notably, Ma *et al.* [12] showed that the public-key algorithm is indispensable; Wang *et al.* [13] introduced a technique integrating "fuzzy-verifier" and "honey-words" to settle the conflict between resisting against offline dictionary attacks and detecting typos timely. In contrast, little attention has been paid to node capture attacks. In most cases, node capture attacks are mentioned as a security threat, but their actual attacking process and consequences are overlooked, and there is a lack of systematic investigation of node capture attacks.

1.1 Node Capture Attacks

Around 2000, Carman *et al.* [14] and a number of researchers [15], [16], [17] pointed out that the adversary can physically acquire the data of some sensor nodes, because they are usually left in unattended or hostile environments and it costs too much to equip them with tamper-resistant hardwares in view of their large-scale development. In 2005, Benenson *et al.* [18] for the first time introduced node capture attacks into remote user authentication schemes. They pointed out that the adversary can compromise some of sensor nodes and carry out a series of subsequent attacks. After that, node capture attacks begin to be accepted as a practical attack against user authentication schemes for WSNs, and many new proposals are designed to resist against this attack.

One notable attempt, initiated by Vaidya *et al.* [19] in 2010, is to identify the weaknesses in previous schemes [20], [21], [22] against node capture attacks and design a new secure version. Unfortunately, this scheme was later shown that it does not provide forward secrecy, and is insecure against smart card loss attacks and node capture attacks. In 2012, Vaidya *et al.* [23] proposed a hash-based scheme which is claimed to be secure against node capture attacks.

However, in 2014, Kim *et al.* [24] pointed out that Vaidya *et al.*'s scheme [23] is unable to resist node capture attacks: if the adversary gets the private key of a sensor node, she can forge the message that is sent by the gateway to users. They then proposed a scheme which is claimed to be secure against node capture attacks and other known attacks. Nevertheless, their security claims were invalidated by Chang *et al.* [25], who showed that a legitimate user can get sensor node's private key and carry out subsequent attacks. Therefore, Chang *et al.* proposed a new enhanced version, and proved their scheme is secure against node capture attacks. However, their scheme was demonstrated by Park *et al.* [26] that it dose not provide forward secrecy if the adversary obtains the private key of a sensor node, and thus it is not secure against node capture attacks. Recently, in 2017, Srinivas *et al.* [27] presented a temporary-certificate-based user authentication

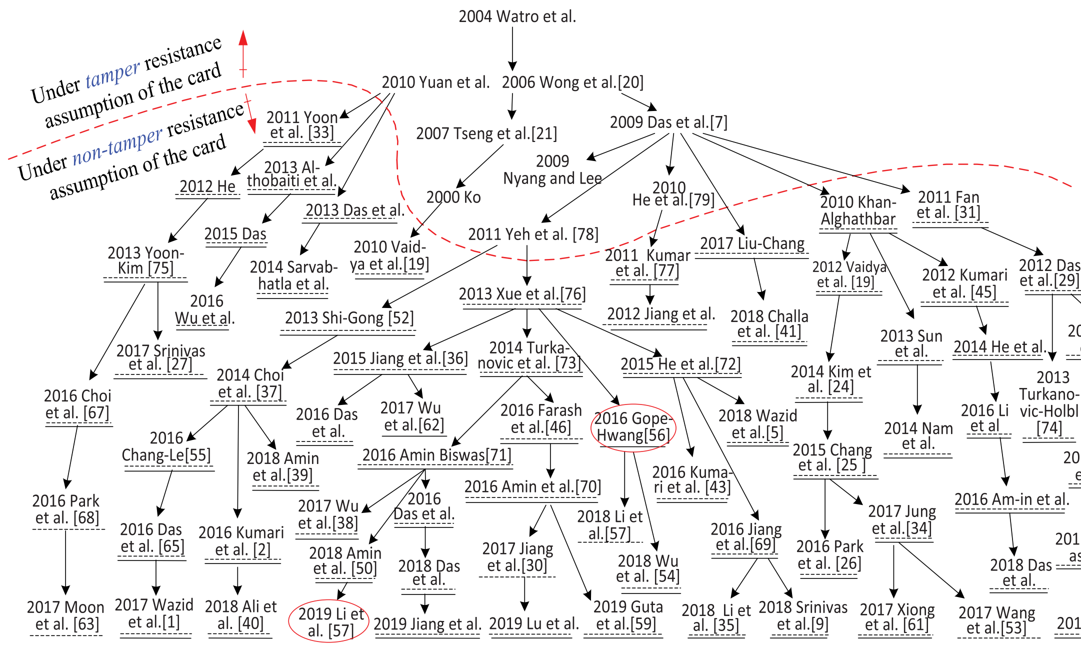


Fig. 2. A brief history of multi-factor user authentication for WSNs. This figure is developed from Fig. 2 of [6]. Most works first present effective attacks on protocol(s) in the parent node, and then propose a new scheme and claim it to be better than previous ones. Schemes underlined with a solid line cannot resist against offline dictionary attacks, and these underlined with a dashed line cannot resist against node capture attacks. Among all schemes, only two of them (those that are circled in the graph) are secure against node capture attacks.

scheme for WSNs with lightweight operations. Particularly, they proved that their scheme is secure against node capture attacks. Later, Wang *et al.* [6] revealed that in Srinivas *et al.*'s scheme [27], once a sensor node is compromised, the adversary is able to compute previous session keys that are associated to this sensor node. Thus, the scheme in [27] is unable to resist node capture attacks again.

As said above, node capture attacks have been considered as a practical attack against user authentication schemes for WSNs. More and more schemes take the resistance to node capture attacks as an attribute that should be satisfied [6], [23], [27], but most schemes still suffer from this threat. Moreover, when assessing the security of multi-factor user authentication schemes for WSNs, node capture attacks are usually included in the criterion "resistance to known attacks" (see [6], [28]). In a nutshell, The harmfulness of node capture attacks have not been well recognized and a systematic investigation is still lacking.

1.2 Motivations and Contributions

Generally, sensor nodes are deployed in unattended or hostile environments, and its large-scale deployment makes it too costly to equip them with tamper-resistant hardwares. Hence, sensor nodes are susceptible to be captured by adversaries, resulting in typical node capture attacks in user authentication schemes for WSNs.

1) Although some recent work takes into account of node capture attacks, as mentioned above, most of them still cannot resist against node capture attacks, and they are caught in a "break-fix-break-fix" circle. The main reason for this undesirable situation is a lack of systematic investigation on node capture attacks.
2) Moreover, the damaging effects of node capture attacks are underestimated. According to our observation and examples in later sections, besides triggering the leakage of previous session keys [6], node capture attacks may enable adversaries to trace user activities, impersonate users, manipulate not only the compromised sensor nodes but also other nodes, and even break the security of the entire system.

In all, node capture attacks have become one of the most urgent and prevalent issues to be addressed in the design of a secure user authentication scheme for WSNs, and they would have a huge impact on the security of user authentication schemes. Understanding node capture attacks and summarizing their causes and consequences can help to design a secure authentication scheme that can resist against this kind of attack, which motivates us to conduct a systematic investigation on node capture attacks.

As far as we know, this is the first in-depth exploration on node capture attacks in the field of user authentication schemes for WSNs. Towards our goal, we first define the adversary model based on Wang *et al.*'s work [6]. Then, we put forward a detailed and thorough evaluation criteria for multi-factor user authentication schemes for WSNs. We achieve this by combining the merits of the widely accepted evaluation criteria [6], [13] and including the effects of node capture attacks. Note that, unlike [6], [13], where they include the resistance to node capture attacks in the criterion C5 "resistance to known attacks", we propose a separate criterion "resistance to node capture attacks". This additional criterion is indispensable to understand and evaluate the security of multi-factor user authentication schemes for WSNs due to the prevalent feature and damaging effects of node capture attacks.

Then, with intensive experience on analyzing about ninety user authentication schemes for WSNs, we figure out the various causes and consequences of node capture attacks, and classify them into ten types in terms of the attack targets, adversary's capabilities and vulnerabilities exploited. We explain each type of attack through examining typical vulnerable schemes, and propose corresponding countermeasures. For example, in Fan *et al.*'s scheme [29], due to the inappropriate distribution of sensor nodes' private keys, all sensor nodes share a same private key with the gateway. We show that an adversary who compromises the sensor node $\mathsf{SN_j}$, can obtain the private keys of all sensor nodes, resulting to sensor node impersonation threat. To deal with this attack, we recommend to use $\mathsf{h(ID_{SN_j}||x)}$ as $\mathsf{SN_j}$'s private key, where x is a long-term secret key, $\mathsf{ID_{SN_j}}$ is $\mathsf{SN_j}$'s identity.

Finally, according to our taxonomy of node capture attacks, we naturally improve our evaluation criteria for multi-factor user authentication schemes for WSNs by expanding the criterion "resistance to node capture attacks" into ten types. We then perform a large-scale assessment of 61 multi-factor user authentication schemes for WSNs under our expanded criteria set. Among those schemes, only two are secure against node capture attacks, indicating the difficulty in designing node-capture-attack resistant user authentication schemes for WSNs along the way. Fortunately, our work provides a better understanding of node capture attacks, and we believe that this work would facilitate the design of secure user authentication schemes for WSNs that is resistant to node capture attacks. In brief, our contributions are summarized as follows:

1) We investigate the root causes and consequences of node capture attacks against user authentication schemes for WSNs, and classify them into ten different types in terms of the attack targets, adversary capabilities and vulnerabilities exploited. As far as we know, we are the first to provide a taxonomy of node capture attacks.
2) We elaborate on each type of node capture attacks through examining a corresponding typical vulnerable scheme, and propose corresponding countermeasures.
3) Finally, based on our taxonomy of node capture attacks, we extend our evaluation criteria, and perform a large-scale assessment of 61 user authentication schemes for WSNs under the expanded criteria.

1.3 Paper Organization

The remaining sections are organized as follows. In Section 2, we describe the adversary model, evaluation criteria and notions used in the paper. Section 3 presents a taxonomy of node capture attacks. Section 4 explains each type of node capture attacks using several typical schemes. The countermeasures are given in Section 5. Section 6 gives a large-scale measurement of 61 representative authentication schemes under our extended evaluation criteria. The summary of this paper is given in Section 7.

TABLE 1
Notations and Abbreviations

Symbol	Description	Symbol	Description
U	user	SN	sensor node
U_i	the i^{th} user	SN_j	the j^{th} sensor node
GWN	gateway node	SC_i	U_i's smart card/device
$\mathcal{A}$	the adversary	x	GWN's long-term secret key
$\mathsf{ID_i}$	identity of U_i	PW_i	password of U_i
$\mathsf{ID}_{\mathsf{SN}_j}$	identity of SN_j	SK	session key*
$\mathsf{X}_{\mathsf{U}_i}$	secret key of U_i	$\mathsf{X}_{\mathsf{SN}_j}$	secret key of SN_j
$\oplus$	bitwise XOR operation	$\|$	concatenation operation

* :*throughout the paper, the session key is between the user and sensor node.*

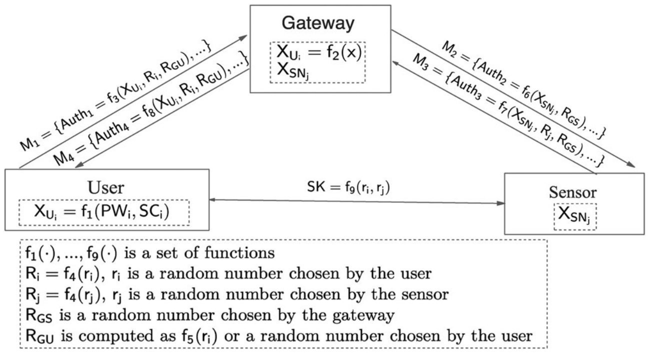


Fig. 3. Authentication processes.

2 Adversary Model, Evaluation Criteria, and Model of Authentication Process

In this section, we first introduce some notations and a standard model of authentication process for public-key based multi-factor user authentication schemes for WSNs, then define the adversary model and evaluation criteria based on widely accepted frameworks.

2.1 A Generic Model of Authentication Process

Our notations and abbreviations are illustrated in Table 1, and the standard model of authentication process for WSNs is shown in Fig. 3. Note that, this model is recommended by Wang *et al.* [6], because other models for single-gateway WSNs have some inherent weaknesses.

In a user authentication protocol, there are three parties: the gateway node GWN, users, and sensor nodes. GWN possesses a secret key x, which is known as the long-term secret key. It is assumed that this key x is well protected and cannot be extracted from GWN's database.[1] For a user U_i, it owns an identity-password pair $(\mathsf{ID}_i, \mathsf{PW}_i)$. When U_i requests to join the network in the registration phase, it interacts with GWN and obtains a smart card/device, which contains information SC_i. After the interaction, U_i and GWN share a secret key $\mathsf{X}_{\mathsf{U}_i}$, which can be computed by $\mathsf{f}_1(\mathsf{PW}_i, \mathsf{SC}_i)$ at the user side and by $\mathsf{f}_2(x)$ at the gateway side. GWN also registers sensor nodes to the network and computes a secret key $\mathsf{X}_{\mathsf{SN}_j}$ as SN_j's private key.

When U_i wants to access data in SN_j, it needs to authenticate itself to SN_j via the help of GWN. To begin with, it first chooses a random parameter r_i and computes $\mathsf{R}_i = \mathsf{f}_4(\mathsf{r}_i)$, $\mathsf{R_{GU}} = \mathsf{f}_5(\mathsf{r}_i)$. It then sends an access request $\mathsf{M_1} = \{\mathsf{Auth_1} = \mathsf{f_3}(\mathsf{X_{U_i}}, \mathsf{R_i}, \mathsf{R_{GU}}), \ldots\}$ to GWN. Note that $\mathsf{M_1}$ always contains the information that helps to compute $\mathsf{Auth_1}$ if the user is honest. Upon receiving M_1, GWN checks the validity of M_1 by checking whether its computed Auth'_1 is equal to $\mathsf{Auth_1}$. If it is valid, GWN chooses a random parameter $\mathsf{R_{GS}}$, computes $\mathsf{M_2} = \{\mathsf{Auth_2} = \mathsf{f_6}(\mathsf{X_{SN_j}}, \mathsf{R_{GS}}), \ldots\}$ and then sends M_2 to SN_j. After receiving M_2, SN_j checks its validity first, chooses a random parameter r_j if valid, and then replies back $\mathsf{M_3} = \{\mathsf{Auth_3} = \mathsf{f_7}(\mathsf{X_{SN_j}}, \mathsf{R_j}, \mathsf{R_{GS}}), \ldots\}$ with $\mathsf{R}_j = \mathsf{f}_4(\mathsf{r}_j)$. With $\mathsf{M_3}$, GWN checks its validity and if valid it replies $\mathsf{M_4} = \{\mathsf{Auth_4} = \mathsf{f_8}(\mathsf{X_{U_i}}, \mathsf{R_i}, \mathsf{R_{GU}}), \ldots\}$ to U_i. If the authentication phase is successful, U_i and SN_j will agree on a secret session key $\mathsf{SK} = \mathsf{f}_9(\mathsf{r}_i, \mathsf{r}_j)$. Among the notations mentioned above, we have the following definitions:

1) We name parameters $\mathsf{Auth_1}$, $\mathsf{Auth_2}$, $\mathsf{Auth_3}$ and $\mathsf{Auth_4}$ used for verifying the validity of participants as *verification parameter*, denoted as VP.
2) The shared secret key $\mathsf{X}_{\mathsf{U}_i}$ is named as *fixed unique secret parameter* between GWN and U_i, denoted as $\mathsf{FUSP_{G/U}}$. Similarly, $\mathsf{X}_{\mathsf{SN}_j}$ is named as *fixed unique secret parameter* between GWN and SN_j, denoted as $\mathsf{FUSP_{G/S}}$.
3) The shared secret parameter $\mathsf{R_{GU}}$ is named as *temporary unique secret parameter* between GWN and U_i, denoted as $\mathsf{TUSP_{G/U}}$. $\mathsf{R_{GS}}$ is named as *temporary unique secret parameter* between GWN and SN_j, denoted as $\mathsf{TUSP_{G/S}}$.
4) Parameter r_i that is chosen by U_i and critical in computing session keys is named as *SK-U-critical parameter*, denoted as $\mathsf{CP_{SK/U}}$. Parameter r_j that is chosen by SN_j and essential in the computation of session keys is named as *SK-S-critical parameter*, denoted as $\mathsf{CP_{SK/S}}$.

Within a user authentication protocol, if an adversary is able to obtain any of: PW_i for some i, $\mathsf{X}_{\mathsf{SN}_j}$ for some j, or the long-term secret key x, we say that the adversary can fully impersonate their victim ($\mathsf{U}_i, \mathsf{SN}_j$ or GWN, respectively). Such an attack is called a *complete impersonation attack*. On the other hand, if the adversary has no such secret information and can only try to manipulate some parties' messages to cheat other parties, we name such an attack as an *incomplete impersonation attack*.

2.2 Adversary Model and Evaluation Criteria

As the security of a cryptographic scheme cannot be properly evaluated if the adversary model or evaluation criteria is not well defined, we now describe the adversary model and evaluation criteria, tailored to multi-factor user authentication protocols for WSNs in the single-gateway setting.

Our adversary model is adapted from the one in [6] and is defined in Table 2. Note that Wang *et al.*'s criteria set [6] only considers the two-factor authentication scenario. Therefore, we adjust C3 of [6] so that it captures the three-factor scenario considered in this work. Also, we remove the adversary's ability in multi-gateway setting in C7 as we only consider the single-gateway environment.

Our evaluation criteria is adapted from the state-of-the art evaluation frameworks [6], [13] and the traditional one [51], and it is illustrated in Table 3. More specifically, following [51], we divide the criteria into two levels: the ideal attributes and security requirements. The former deals with various attributes that an ideal user authentication scheme should provide, and focuses on the usability of the

1. When assessing forward secrecy, this key can be extracted

TABLE 2
Capabilities of the Adversary[0]

C1	$\mathcal{A}$ can control messages transmitted among U, SN and GWN.
C2	$\mathcal{A}$ can offline enumerate all items in the Caresian product of identity space and password space $\mathcal{D}_{id} \times \mathcal{D}_{pw}$ within polynomial time, or get U's identity only when evaluating the scheme's security.
C3[1]	To a n-factor (n = 2 or 3) user authentication scheme, $\mathcal{A}$ can compromise following $n-1$ factors: (1) password; (2) data in smart card; (3) biometric.
C4	$\mathcal{A}$ can acquire previous session keys between U and SN.
C5	$\mathcal{A}$ can learn GWN's secret key(s) when assessing the system's eventual failure.
C6	$\mathcal{A}$ can break some SN, i.e. extracting the sensitive data stored in SN, and controlling the broken sensor node to join the communication of U and GWN.
C7[2]	$\mathcal{A}$ may register to be a legitimate user. Only when assessing the security of the users' passwords in the registration phase, $\mathcal{A}$ can also be the administrator of the gateway.

[0]: *The seven capabilities are not all necessary to Table 4 where C4 and C5 are not mentioned, because these two capabilities have no inherent relevance to the taxonomy of node capture attacks. However, we include all seven capabilities here for completeness.*
[1]: *Note that C3 of Wang's criteria [6] is suitable for the two-factor user authentication schemes, so we add the three-factor condition in C3 to make the model apply to multi-factor user authentication schemes.*
[2]: *Since the multi-gateway environment is not our focus, we omit the part about it. Furthermore, we highlight the security threat of the administrator of the gateway in the registration phase.*

protocol. The latter specifies requirements that a scheme should satisfy to be served as a secure one. Following [6], [13], we remove redundancies in the criteria of [51] and form our 12 independent criteria. Inspired by [6], [13], we separate node capture attacks from "the known attacks" and propose our criterion S6 (resistance to node capture attacks), taking into account the prevalent features and damaging effects of node capture attacks. More specifically, the reason why we propose an independent criteria for node capture attacks, are consistent with Wang *et al.* [13], where they separate smart card loss attacks from the criterion C5 "resistance to known attacks" due to the destructive effects of smart card loss attacks. Another difference of our criteria from [6], [13], [51] is that we specify the adversary's capabilities for each criterion. For the criteria under ideal attributes, we evaluate them from the functional perspective rather than from the attacking view. For the criteria under security requirements, we specify the adversary's capabilities in Table 3.

Remark 1. From the above comparison, we can see that the most important difference between our criteria and existing criteria [6], [13], [51] is that our criteria proposes a separate criterion S6 "resistance to node capture attacks", yet this criterion is included in the criterion "resistance to known attacks" in existing criteria. Furthermore, as shown in Section 6, our criteria framework will further divide S6 into ten sub-criteria based on our analysis results of node capture attacks. This difference is the main reason why it seems that our criteria becomes more complex than others. However, we think this complexity is necessary and it will make our criteria more concrete and decidable to be employed. In these existing criteria [6], [13], [51], the attack scenarios where the adversary simultaneously compromise the victim's smart card and several sensor nodes, cannot be captured. Besides, our criteria framework allows the designers to assess the scheme more objectively and easily. For example, in previous criteria framework, if the protocol designer wants to assess whether their scheme can satisfy the criterion "resistance to known attacks", she needs to assess whether their scheme can resist to node capture attacks. But how to achieve this? Before our work, they need to try various possible attack scenarios, which either may cost more time to assess or ignore some important attack scenarios. Therefore, following these existing criteria frameworks, it is difficult and tricky for protocol designers to assess whether their schemes can resist against node capture attacks, and they need to make much more efforts. Fortunately, the ten sub-criteria (as shown in Section 6) of our criteria framework provide a structured, actionable and concrete reference for protocol designers to systematically evaluate whether their scheme can resist against node capture attacks.

Remark 2. All authentication schemes are assessed from two aspects: (1) The security and functionality under a widely-accepted criteria framework; (2) The performance, such as computational cost and storage cost. The former captures the security and functionality requirements, and

TABLE 3
Evaluation Criteria

	Short term		Definition in WSNs
Ideal Attributes[†]	D1	Password Friendly*	Users can choose their passwords freely and change them locally.
	D2	Sound Repairability	Dynamic sensor node addition and smart card revocation are provided.
	D3	Key Agreement	After authentication, a session key is agreed between the user and the sensor node.
	D4	No clock synchronization	Users, the gateway and sensor nodes need not to synchronize their time clock.
	D5	Mutual Authentication	Users, sensor nodes and gateway should authenticate each other.
	D6	No Password Verifier table	GWN and SN shall not store any password-related value.
Security Requirements	S1	User Anonymity	$\mathcal{A}$ with capabilities C1, C2 and C4 cannot compute users' identity or trace users' activities.
	S2	No Password Exposure	$\mathcal{A}$ with capabilities C2, C3 and C7 cannot acquire users' passwords in the registration phase.
	S3	Forward Secrecy	$\mathcal{A}$ with capabilities C1-C5 and C7 compromises any number of the parties, the session key between users and sensor nodes still cannot be computed.
	S4	Resistance to Known Attacks	$\mathcal{A}$ with all capabilities (except for breaking the victim's smart card and some of sensor nodes) can resist against impersonation attacks, offline guessing attacks, de-synchronization attacks, replay attacks, parallel attacks, key control, stolen verifier-attacks, unknown key share attacks and known key attacks.
	S5	Resistance to Smart Card Loss Attacks	$\mathcal{A}$ with all capabilities besides C6 cannot carry out any attacks with the help of the victim's smart card, such as guessing or changing the password, and impersonate the victim.
	S6	Resistance to Node Capture Attacks	$\mathcal{A}$ with all capabilities cannot conduct attacks with the help of compromised sensor nodes.

[†]:*An ideal attribute is assessed from the functional perspective rather than an attack.*
[⋆] :*The criterion "Timely Typo Detection" in [6] is included in D1 here, as a scheme providing local-change-password can timely detect typos too.*
Note that, we say that $\mathcal{A}$ breaks S5 and S6 only when $\mathcal{A}$ conducts the attack with the help of compromised smart card and sensor node, respectively.

TABLE 4
A Taxonomy of Node Capture Attacks[1]

Type	Attack Target	$\mathcal{A}$'s Capabilities					Vulnerabilities Exploited	Attack Consequences	Attack Scale[2]	Refer.	Vulnerable Schemes
		C7	C6′	C3′	C2	C1′					
I	Session key	×	×	×	×	×	the issue of forward secrecy	get previous SK of SN_j	(a)→(c)	Sec. 4.1	[2], [30]
II		×	×	×	×	×	1.inappropriate distribution of SN's secret key, and, 2.the issue of forward secrecy	get previous SK of SN_m	(a)→(i)→(d)	Sec. 4.2	[29], [31] [23], [32]
III	Users	×	×	×	×	×	insecure transmission of U's unique secret parameter	get $FUSP_{G/U}$	(a)→(b)→(e)→(f)	Sec. 4.3	[11], [33]
IV		×	×	√	√	×	same to offline dictionary attacks in distributed system no (or incorrect) deployment public-key algorithm	get users' password	(a)→(b)→(e)→(f)	Sec. 4.4	[30], [34] [35], [36]
V		×	D	×	×	×	1.insecure transmission (eg."XOR") of $ID_i/h(ID_i)$ or, 2.unreasonable design intent, or 3.problems caused by temporary certificates	break user anonymity (track U or compute ID_i)	(a)→(b)→(e)	Sec. 4.5	[37], [38] [34], [39] [30], [40]
VI	Sensor node	×	×	×	×	×	1.inappropriate distribution of SN's secret key, or, 2.insecure transmission (eg."XOR") of SN's secret key	get SN_m's secret key	(a)→(i)→(d)	Sec. 4.6	[29], [33] [24], [41]
VII		×	×	D	×	√	1.exposure of U's unique secret parameters, and 2.GWN's inefficient authentication to SN[3]	impersonate SN_m	(a)→(b)→(e)→(f)	Sec. 4.7	[40], [42] [43], [44]
VIII		×	√	×	×	√	1.U's login request fails to identify the target SN, and, 2.U's login request first sends to SN_j[4]	impersonate SN_m	(a)→(j)→(g)	Sec. 4.8	[45], [46] [47], [48]
IX	Gateway	√	×	×	×	×	insecure transmission of GWN's secret key $x/h(x)$	get GWN's secret key	(a)→(h)	Sec. 4.9	[33], [49]
X	Availability	×	D	D	×	√	1.U fails to verify $CP_{SK/SN}$, and, 2.insecure transmission $CP_{SK/U}$	modify session key witho-ut noticed by participates	(a)→(c), or (a)→(b)→(e)→(f)	Sec. 4.10	[38], [50] [34], [39]

- [1] *In this table, we assume the adversary has broken the sensor node* SN_j, U_i *is a legitimate user and welling to collude with* $\mathcal{A}$, *as well as a victim that* $\mathcal{A}$ *tries to attack.* SN_m $(m \neq j)$ *denotes the sensor nodes that* $\mathcal{A}$ *attempts to attack.*
- *C6′, C3′ and C1′ all are a part of the capabilities C6, C3 and C1, respectively. C6′: here we want to emphasize that* $\mathcal{A}$ *acts as* SN_j *to actively participates in the communication. C3′: here C3′ refers to that* $\mathcal{A}$ *extracts the data in* U_i*'s card (and gets the biometrics). C1′: here C1′ refers to that* $\mathcal{A}$ *modifies and sends message to the participants (U/SN/GWN). Since C4 and C5 have no influence to the classification, they are not listed.*
- *×: The capability is not required. √: the capability is required. D: whether the capability is required depends on specific attack scenario.*
- [2]*: As shown in Fig. 4, it describes the status of the affected entity with the increase in the number of attacks.*
- [3]*: It usually happens in the model (b) of Fig. 5.*
- [4]*: It usually happens in the model (c) of Fig. 5.*

is continuing to be a hot and hard topic and has led to intense research, see [6], [13], [51]. The latter is specific and can well capture the dynamic nature of WSNs. Like existing criteria frameworks, the target of our criteria is to assess the security and functionality of authentication schemes. When WSNs become larger, the factors that affect the security, such as the capabilities to compromise the victim's smart card, some of sensor nodes and long-term secret key, will not change. Therefore, the scale of WSNs has little impact on the security of authentication schemes. As for the functionality, the criterion "Sound Repairability", which requires the scheme to support dynamic sensor node addition, is an attribute to support the dynamic nature of WSNs. Therefore, when WSNs become larger, existing criteria frameworks and ours are still workable.

3 A Taxonomy of Node Capture Attacks

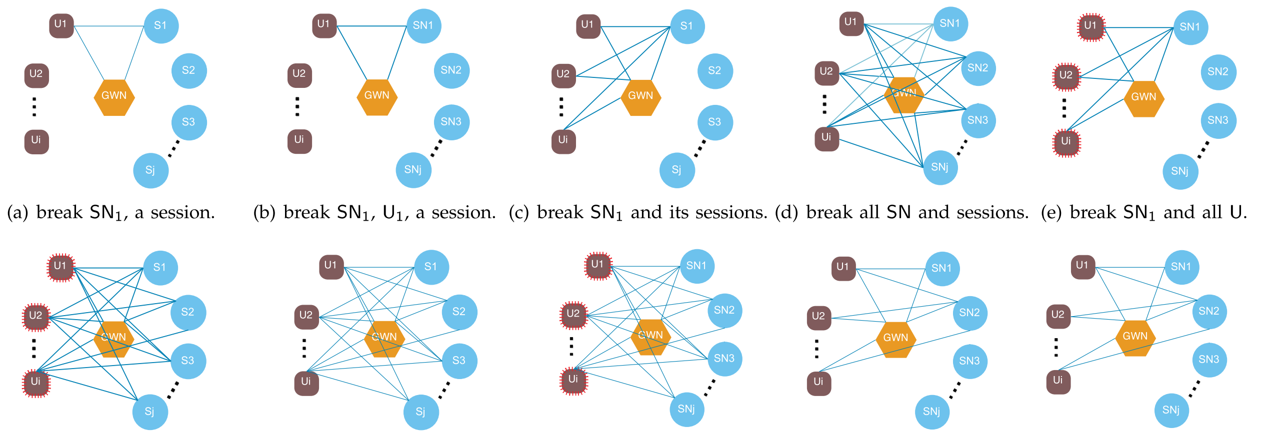
Based on the analysis of about 90 authentication protocols for WSNs, we investigate the causes and consequences of node capture attacks, and we classify them into ten different types (see Table 4) in terms of the attack targets, adversary's capabilities and vulnerabilities exploited. As shown in Table 4, the adversary $\mathcal{A}$ can achieve different attack consequences and attack scale[2] in terms of different attack targets, adversary's capabilities and vulnerabilities exploited. The attack targets can be divided into five categories: the session keys, the users, the sensor nodes, the gateway and the availability. The vulnerabilities exploited include insecure parameter transmission, inappropriate parameter distribution, unreasonable design intent, inefficient verification, the issue of offline dictionary attacks and forward secrecy.

2. Since it is more easy to understand the attack scale with specific examples, we explain it in Section 4.

Type I and Type II in Table 4 depict the attacks where $\mathcal{A}$ breaks the security of session keys with the help of the private key X_{SN_j} of compromised sensor node SN_j. The difference between them is that: in Type I, $\mathcal{A}$ can only calculate previous session keys between the *compromised sensor node* SN_j and all users; in Type II, $\mathcal{A}$ can calculate previous session keys between *all sensor nodes* and all users, it is more destructive. The root cause of Type I is essentially consistent with that of forward secrecy. As for Type II, besides the issue of forward secrecy, the inappropriate distribution of SN's private key is its another cause.

Type III∼Type V in Table 4 represent the scenarios where the adversary with X_{SN_j} compromises the security of users. Both Type III and Type IV can be regarded as a user impersonation attack. In Type III, the adversary gets the victim's fixed unique secret parameter $FUSP_{G/U}$ (X_{U_i} in Fig. 3) to impersonate U_i, due to the insecure transmission (such as "XOR" operation) of user unique secret parameters. In Type IV, the adversary with X_{SN_j} and who additionally gets the data stored in a victim's smart card (and the biometric) and can enumerate the items in the space of password and identity, is able to obtain users' passwords, and then acts as U_i to engage in the conversations. This attack is an outcome of no (or incorrect) public-key algorithm deployment. Its failure reason is as same as offline dictionary attacks [12].

If the adversary $\mathcal{A}$ cannot impersonate the user, then she may try to break the victim's privacy, and this is the situation in Type V. It contains two cases. In the first case, $\mathcal{A}$ can trace users' activities, but cannot compute their identities. It usually occurs in a temporary-certificate-based authentication protocol where $\mathcal{A}$ can trace the victim by manipulating the broken sensor node to seek the link parameter, such as TID_i in [37]. In the second case, $\mathcal{A}$ can compute users' identities ID_i successfully, because they are transmitted with simple "XOR" operation or designed to be known to sensor nodes



(a) break SN_1, a session. (b) break SN_1, U_1, a session. (c) break SN_1 and its sessions. (d) break all SN and sessions. (e) break SN_1 and all U.

(f) break SN_1, all U and sessions. (g) break SN_1 and all sessions. (h) break the whole system. (i) break SN_1 and SN_2. (j) break SN_1, some sessions.

Fig. 4. The scale of attacks. We use the attack diagram to show the affected parties and sessions. Those marked in red circle represent that some of its parameters have been obtained by the adversary. The affected sessions are marked by a blue line.

(e.g., simply transmit ID_i to sensor nodes in plaintext), and we call this flaw an unreasonable design intent.

In Type VI~Type VIII in Table 4, once the adversary $\mathcal{A}$ compromises the sensor node SN_j and gets X_{SN_j}, then she also can compromise other sensor nodes SN_m ($m \neq j$) in different ways. In Type VI, $\mathcal{A}$ with X_{SN_j} exploits the insecure transmission or distribution of the private key of sensor nodes SN_m ($m \neq j$) to obtain the private key and impersonate SN_m. In Type VII, due to the insecure transmission of users' unique secret parameters and GWN's failure in authenticating SN (usually happen in the communication model of (b) in Fig. 5), $\mathcal{A}$ with X_{SN_j} then can impersonate SN_m to users. In Type VIII, since the users' login requests are first sent to sensor nodes without explicitly designating the target sensor node SN_m (usually happens in the communication model of (c) in Fig. 5), the adversary with X_{SN_j} then can intercept the login request sent to SN_m, and act as SN_j to respond the request as the process of the original protocol without being noticed by users. After this attack, the users think a session key is agreed with SN_m, but actually with SN_j (i.e., the adversary). Among the three attacks, we say that only Type VI where the adversary *becomes* SN_m achieves *complete impersonation*. $\mathcal{A}$ in Type VII and VIII tries to disguise as much as possible to deceive U_i and GWN, so only achieves *incomplete impersonation*.

In Type IX, the adversary who registers as or colludes with a legitimate U_i and exploits the weakness in the insecure transmission of GWN's long-term secret key, can get the secret key. It makes whole system completely insecure, because GWN's long-term secret key is employed to compute all secret information of users and sensor nodes.

The last attack Type X is very common in authentication schemes based on our analysis, but it receives little notice. In Type X, the adversary $\mathcal{A}$ with X_{SN_j} can modify session keys between users and sensor nodes. Furthermore, it is a progressive attack. If U_i fails to verify part of SK controlled by SN_j, i.e., $CP_{SK/S}$, then $\mathcal{A}$ with SN_j's private key can tamper the respond message from SN_j/GWN to users (i.e., message M_4 in Fig. 3), and makes legitimate participants (i.e., *all users and* SN_j) unable to share the same session key, meanwhile the participants authenticate to each other successfully. If besides the problem above, the part of SK controlled by U_i ($CP_{SK/U}$) is also transmitted insecurely, then $\mathcal{A}$ can modify session keys between U_i *and all sensor nodes*. Type X not only causes usability problems where legitimate parties cannot share a same session key and thus cannot correctly decrypt their interaction messages, but also enable the adversary to compute the same session key with U_i.

4 Examples of the Ten Types of Node Capture Attacks

To better understand the ten different types of attacks in Table 4, we explain them in detail by using several typical user authentication protocols for WSNs. Note that, to save space, we do not review the original protocols, and we retain the symbols and notations of the original protocols even though they are not the same as those in Table 1.

4.1 Node Capture Attack Type I

Type I depicts a practical attack where the adversary with the private key X_{SN_j} of sensor node SN_j can compute the session keys between SN_j and all users. The attack cause of Type I is the same as that of forward secrecy. This section uses Kumari *et al.*'s scheme [2] to explain this attack.

- *Adversary's Capability*:
  (1) "C1". Eavesdrop the message between GWN and SN_j in authentication phase to get $\{A5_i, A6_i, C2_i\}$.
  (2) "C6". Get SN_j's private key SIX_k.
- *Attack Target*: session key.
- *Attack Consequence*: compute SN_j's previous session keys.
- *Attack Steps*:

Step 1. Compute $(RU_i^{**}||RG_j^*||TS2_i^*) = A5_i \oplus SIX_k$.
Step 2. Compute $A1_i^* = A6_i \oplus h(SID_k||h(RG_j^*)||RU_i^{**})$.
Step 3. Compute $(RS_k^*||TS3_i^*) = C2_i \oplus RG_j^*$.
Step 4. Compute $SK_i = h(A1_i^*||RU_i^{**}||RG_j^*||RS_k^*)$.

- *Time Complexity*: $O(3T_H)$, where T_H denotes the running time of hash operation. Some lightweight operations like XOR and || are omitted.
- *The Scale of the Attack*: (a)→(c) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ compromises the

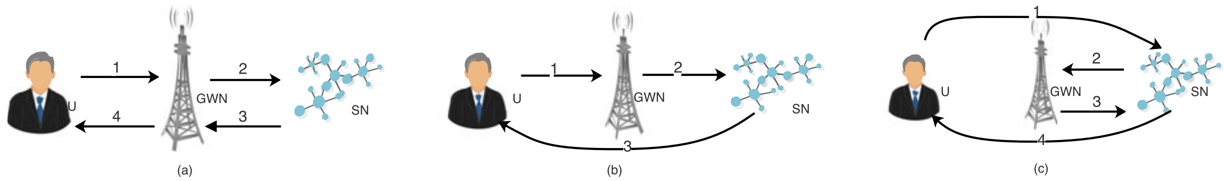


Fig. 5. Communication models of user authentication schemes for WSNs [6].

sensor node SN_1 (marked in red circle as shown in (a) of Fig. 4), then exploits messages among GWN, SN_1 and U_1, $\mathcal{A}$ can compute previous session keys between U_1 and SN_1 as above. Similarly, $\mathcal{A}$ can compute previous session keys between U_2 and SN_1. When the number of the attacks is large enough, $\mathcal{A}$ can compute previous session keys between SN_1 and all users as shown in (c) of Fig. 4.

4.2 Node Capture Attack Type II

As we mentioned above, the issue of forward secrecy and inappropriate distribution of sensor nodes' private keys result in Type II. In this section, we take Fan *et al.*'s scheme [29] as an example to describe node capture attack Type II. Note that all sensor nodes and the gateway share the same secret parameter S_k in Fan *et al.*'s scheme [29].

- *Adversary's Capability*:
  (1) "C1". Eavesdrop the message between GWN and SN_j in authentication phase to get K.
  (2) "C6". Get SN_j's private key S_k.
- *Attack Target*: session key.
- *Attack Consequence*: compute previous session keys between all sensor nodes and all users.
- *Attack Steps*:

Step 1. Compute $Key = h(S_k||K)$.

- *Time Complexity*: $O(T_H)$.
- *The Scale of the Attack*: (a)→(i)→(d) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ gets SN_1's private key (marked in red circle as shown in (a) of Fig. 4), then exploits messages among GWN, SN_1 and U_1, $\mathcal{A}$ can compute session keys between U_1 and SN_1 as above. Since $\mathcal{A}$ also gets all SN's private key in the first attack, $\mathcal{A}$ then can compute session keys between U and SN_1 as shown in (i) of Fig. 4. When the number of the attacks is large enough, $\mathcal{A}$ can compute previous session keys between all users and all sensor nodes as shown in (d) of Fig. 4.

From the above attacks, we can see that securely distributing the private key of the sensor nodes is fundamental to the whole security, it also decides the authentication process. Inappropriate private key distribution can cause sensor node impersonation and session key leakage.

4.3 Node Capture Attack Type III

Type III utilizes the vulnerability of insecure transmission of user's fixed unique secret parameter $FUSP_{G/U}$ to obtain the necessary information to impersonate the user. This section introduces Type III via Li *et al.*'s scheme [34] as follows:

- *Adversary's Capability*:
  (1) "C1". Eavesdrop the authentication message between GWN and SN_j to get M_8 and M_9, and the message between GWN and U_i to get M_{14} in the session of SN_j.
  (2) "C6". Get SN_j's secret key K_{GWN-S}.
- *Attack Target*: the users.
- *Attack Consequence*: get $FUSP_{G/U}$ to impersonate U_i.
- *Attack Steps*:

Step 1. Compute $ID_i = M_8 \oplus K_{GWN-S}$.
Step 2. Compute $r_g = h(ID_i||K_{GWN-S} \oplus M_9)$.
Step 3. Compute $M_1 = M_{14} \oplus r_g$, once acquires M_1, $\mathcal{A}$ has the ability to forge the message sent by U_i to spoof GWN and sensor nodes as follows:
Step 4. Generate $r_i^{\mathcal{A}}$ and $s^{\mathcal{A}}$.
Step 5. Compute $M_2^{\mathcal{A}} = s^{\mathcal{A}}P$.
Step 6. Compute $M_3^{\mathcal{A}} = s^{\mathcal{A}}X$, note that X is a public parameter and can be easily gotten.
Step 7. Compute $M_4^{\mathcal{A}} = ID_i \oplus M_3^{\mathcal{A}}$.
Step 8. Compute $M_5^{\mathcal{A}} = M_1 \oplus r_i^{\mathcal{A}}$.
Step 9. Compute $M_6^{\mathcal{A}} = h(ID_i||r_i^{\mathcal{A}}) \oplus SID_m$ (m can be any valid number).
Step 10. Compute $M_7^{\mathcal{A}} = h(M_1||SID_m||M_3^{\mathcal{A}}||r_i^{\mathcal{A}})$.
Step 11. Send $\{M_2^{\mathcal{A}}, M_4^{\mathcal{A}}, M_5^{\mathcal{A}}, M_6^{\mathcal{A}}, M_7^{\mathcal{A}}\}$, then GWN and SN_m will believe the legitimacy of $\mathcal{A}$ and they will build a shared session key successfully. The following procedures are similar to original scheme.

- *Time Complexity*: near to a legitimate user.
- *The Scale of the Attack*: (a)→(b)→(e)→(f) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ gets SN_1's private key (marked in red circle as shown in (a) of Fig. 4), then exploits messages among GWN, SN_1 and U_1, $\mathcal{A}$ can computes U_1's $FUSP_{G/U}$ as above (marked in red circle as shown in (b) of Fig. 4). Similarly, $\mathcal{A}$ can get U_2's $FUSP_{G/U}$. With the increase in number of the attacks, $\mathcal{A}$ can get all users' $FUSP_{G/U}$ as above (shown in (e) of Fig. 4). With users' $FUSP_{G/U}$, $\mathcal{A}$ finally can impersonate all users to all sensor nodes, as shown in (f) of Fig. 4.

4.4 Node Capture Attack Type IV

Type IV is an *complete impersonation* where $\mathcal{A}$ with the private key of SN_j can get victim's password and identity via offline dictionary attacks. Generally, offline dictionary attacks occur when $\mathcal{A}$ can find a *verification parameter* VP to test the correctness of guessed value and is one of the most common attacks in user authentication schemes. Though as we mentioned in Sec. 1, Wang *et al.* [52] introduce the public-key technique to resist against offline dictionary attacks, the situation in WSNs is a little bit different where $\mathcal{A}$ can obtain some sensor nodes's private key to gain many advantages to conduct such an attack. Furthermore, when analyzing offline dictionary attacks, most protocol designers focus on the first two message flows, while pay little attention to subsequent flows sent by the gateway. This well-explains why such an attack is ignored by Jiang *et al.* [30] when they already have known the way to apply public-key algorithm to withstand this attack.[3]

– *Adversary's Capability*:
  (1) "C1". Intercept the message between GWN and SN_j to get M_5, M_7, T_2 in authentication phase.
  (4) "C2". Offline enumerate all items in the space of password and identity.
  (3) "C3". Obtain biometrics fng_i and f_i in smart card.
  (4) "C6". Get SN_j's private key X_j.
– *Attack Target*: the users.
– *Attack Consequence*: compute the password of U_i, then further impersonate U_i.
– *Attack Steps*:

Step 1. Guess PW_i to be PW_i^* and ID_i to be ID_i^*.
Step 2. Compute $K_i^{**} = M_5 \oplus h(ID_i^*||ID_j||X_j||T_2)$.
Step 3. Compute $K_j' = M_7 \oplus K_i^{**}$.
Step 4. Compute $SK = h(ID_i^*||ID_j||K_i^{**}||K_j')$.
Step 5. Compute $B_i^* = BH(r_i, fng_i)$.
Step 6. Compute $d_i^* = f_i \oplus h(ID_i^*||PW_i^*||B_i^*)$.
Step 7. Compute $M_8^* = h(SK||ID_i^*||d_i^*||K_j)$.
Step 8. Verify PW_i^* and ID_i^* by checking if $M_8^* ?= M_8$.
Step 9. Repeat Step 1 $\sim$ 8 until the correct value of PW_i and ID_i are found.

– *Time Complexity*: $O(|\mathcal{D}_{pw}| * |\mathcal{D}_{id}| * (4T_H + T_B))$, where T_B is the time for biometric-specific operation.
– *The Scale of the Attack*: (a)→(b)→(e)→(f) as shown in Fig. 4. The attack evolution is similar to that of Type III.

4.5 Node Capture Attack Type V

Type V contains two kinds basic attacks: (1) track users, or. (2) compute users' identity. Normally, there are three conditions to led to this attack. 1) $\mathcal{A}$ with SN_j's private key may exploit protocol's unreasonable design intent to get users' identity, such as Li *et al.*'s scheme [34] where SN is designed to get ID_i. 2) $\mathcal{A}$ also may make use of the vulnerability in a temporary-certificate-based scheme to track users, such as Wu *et al.*'s scheme [37], where $\mathcal{A}$ can compute TID_i^{new} via eavesdropped D_{10}, T_4 and computed r_u. Once with TID_i^{new}, $\mathcal{A}$ is able to track U_i's next message to learn users' habits and preferences for business purpose. A limitation in the attack on Wu *et al.*'s scheme is that $\mathcal{A}$ can only trace activities of U_i just after U_i interacts with SN_j. 3) $\mathcal{A}$ can exploit the insecure transmission of user identity to compute victim's identity, and we take Amin *et al.*'s scheme [38] to explain this attack as follows:

– *Adversary's Capability*:
  (1) "C1". Eavesdrop the message between GWN and SN_j to get $\{N_j, SS_j, V_j, T_2\}$ and K_{ij}, and the message between GWN and U_i to get M_2 during the authentication phase.
  (2) "C6". Get SN_j's private key f_j.
– *Attack Target*: the users.
– *Attack Consequence*: compute U_i's identity ID_i.

3. They transmit user-chosen random number K_i to GWN via a public-key encryption algorithm. So that $\mathcal{A}$ cannot conduct offline dictionary attacks using M_2 (=$h(d_i||L_i||K_i||T_1)$)

– *Attack Steps*:

Step 1. Compute $h(ID_i) = SS_j \oplus h(f_j||T_2)$.
Step 2. Compute $K_i' = V_j \oplus h(ID_i)$.
Step 3. Compute $K_j' = K_{ij} \oplus K_i'$.
Step 4. Compute $SK = h(h(ID_i)||SID_j||K_i'||K_j')$.
Step 5. Compute $ID_i = M_2 \oplus h(SK||K_i)$.

– *Time Complexity*: $O(4T_H)$.
– *The Scale of the Attack*: (a)→(b)→(e) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ gets SN_1's private key (marked in red circle as shown in (a) of Fig. 4), then exploits messages among GWN, SN_1 and U_1, $\mathcal{A}$ can compute U_1's identity as above as shown in (b) of Fig. 4. Similarly, $\mathcal{A}$ can get U_2's identity. When the number of the attacks is large enough, $\mathcal{A}$ can get all users' identity as shown in (e) of Fig. 4.

4.6 Node Capture Attack Type VI

Type VI depicts an attack where the adversary with SN_j's private key X_{SN_j} can acquire SN_m's ($m \neq j$). Two situations will result in this attack: (1) The inappropriate distribution of SN's private key. (2) The insecure transmission of SN's private key. We take Fan *et al.*'s scheme to show the first situation. In Fan *et al.*'scheme, since all the sensor nodes share a same private key, once $\mathcal{A}$ compromises SN_j to get S_k, then the private key of SN_m is exposed too. Then $\mathcal{A}$ can impersonate all sensor nodes with S_k.

We take Dhillon *et al.*'s scheme [41] as an example to show the second situation where $\mathcal{A}$ can learn SN_m' private key due to the insecure transmission of the key:

– *Adversary's Capability*:
  (1) "C1". Eavesdrop the message between GWN and SN_m to get A_m, e_m and TS2, and the message between U_i and GWN to get TS1 in the session of SN_m ($m \neq j$).
  (2) "C6". Get SN_j's private key X_{gn}, note that X_{gn} is a shared secret between GWN and SN.
– *Attack Target*: sensor node.
– *Attack Consequence*: get SN_m's private key.
– *Attack Steps*:

Step 1. Compute $y_m = A_m \oplus H(X_{gn}||TS1||TS2)$.
Step 2. Compute $x_m = y_m \oplus e_m$, note that x_m is a private key for SN_m, thus now $\mathcal{A}$ can impersonate SN_m. Since the interaction processes are the same as original scheme, we omit here.

– *Time Complexity*: $\mathcal{O}(4T_H)$ for getting SN_m's private key.
– *The Scale of the Attack*: (a)→(i)→(d) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ compromises the sensor node SN_1 (marked in red circle as shown in (a) of Fig. 4), then exploits messages among SN_2, U and GWN to get the private key of SN_2 as above. With SN_2's private key , $\mathcal{A}$ can impersonate SN_2 to all users, as shown in (i) of Fig. 4. After enough interactions, $\mathcal{A}$ can get all sensor nodes' private keys and impersonate any sensor nodes to any users, as shown in (d) of Fig. 4.

4.7 Node Capture Attack Type VII

Type VII presents an impersonation attack where the adversary with SN_j's private key gets U's unique secret parameter, and then exploits GWN's inefficient authentication to

SN to impersonate SN_m $(m \neq j)$. It usually occurs in the communication model (b) of Fig. 5. Type VI and Type VII both are about impersonating sensor node, while Type VI is the *complete impersonation*, Type VII is an elaborate camouflage, i.e., "Incomplete Impersonation". We take Kumari *et al.*'s scheme [42] as an example to explain this attack:

- *Adversary's Capability*:
  - (1) "C1". Eavesdrop D_{g^1} in the session between GWN and SN_j. Furthermore, $\mathcal{A}$ joins the session actively, intercepts and modifies messages among participants.
  - (2) "C6". Get SN_j's private key TC_j.
- *Attack Target*: sensor node.
- *Attack Consequence*: get U_i's unique secret parameters, then impersonate SN_m $(m \neq j)$ to U_i.
- *Attack Steps*:

Step 1. Compute $h(ID_i||h(Q_i)) = D_{g^1} \oplus h(TC_j)$.
Step 2. Intercept message to SN_m: $\{D^m_{g^1}, D^m_{g^2}, C^m_g, T^m_g\}$, note that this session is among U_i, SN_m and GWN.
Step 3. Compute $I_3 = D^m_{g^1} \oplus D^m_{g^2} \oplus h(ID_i||h(Q_i))$.
Step 4. Select a random number $K^{\mathcal{A}}_s$, compute $S^{\mathcal{A}}_1 = T_{K^{\mathcal{A}}_s}[h(ID_i||h(Q_i))] \bmod p$.
Step 5. Compute $SK^{\mathcal{A}}_{s-u} = T_{K^{\mathcal{A}}_s}(I_3) \bmod p$.
Step 6. Compute $S^{\mathcal{A}}_2 = h(SK^{\mathcal{A}}_{s-u}||h(ID_i||h(Q_i))||T^{\mathcal{A}}_s)$, where $T^{\mathcal{A}}_s$ is timestamp.
Step 7. Send $\{S^{\mathcal{A}}_1, S^{\mathcal{A}}_2, T^{\mathcal{A}}_s\}$ to U_i, then according to the protocol, U_i will authenticate $\mathcal{A}$ successfully and share session key SK_{u-s} with $\mathcal{A}$.

- *Time Complexity*: close to legitimate SN_m.
- *The Scale of the Attack*: (a)→(b)→(e)→(f) as shown in Fig. 4. In the beginning, $\mathcal{A}$ compromises the sensor node SN_1 (marked in red circle as shown in (a) of Fig. 4), then exploits messages among U_1, SN_1 and GWN to get U_1's $FUSP_{G/U}$ (marked in red circle as shown in (b) of Fig. 4) as above. Similarly, $\mathcal{A}$ can get U_2's $FUSP_{G/U}$. When the number of the attacks is large enough, $\mathcal{A}$ can get all users' $FUSP_{G/U}$ as shown in (e) of Fig. 4, and then impersonate any sensor nodes to communicate with any users as shown in (f) of Fig. 4.

Note that, Type VI and Type VII have an obvious difference in attack consequence: after an attack of Type VII, $\mathcal{A}$ can impersonate *all sensor nodes* to U_i, and after an attack in Section 4.6 (Type VI), $\mathcal{A}$ can impersonate SN_m to *all users*.

4.8 Node Capture Attack Type VIII

Type VIII usually happens where user's login request first sends to a sensor node rather than the gateway such as model (c) of Fig 5. In this case, once the request is not well marked the target sensor node, $\mathcal{A}$ can intercept it and then carry out an impersonation attack. The similar attack can be found in Shi *et al.*'s scheme [53] criticized by Choi *et al.* [36]. In this section, we introduce node capture attack Type VIII via Farash *et al.*'s scheme [45].

- *Adversary's Capability*:
  - (1) "C1". Intercept messages and send messages to GWN.
  - (2) "C6". Control SN_j, i.e., $\mathcal{A}$ gets $x_j, h(X_{GWN}||1)$ and joins the communication among U_i, SN_m and GWN actively.
- *Attack Target*: sensor node.
- *Attack Consequence*: impersonate SN_m.
- *Attack Steps*:

Step 1. Intercept $\{M_1, M_2, M_3, T_1\}$ from U_i to SN_m, note that this session is among U_i, SN_m and GWN.
Step 2. Compute $ESID^{\mathcal{A}}_j = SID^{\mathcal{A}}_j \oplus h(h(X_{GWN}||1)||T^{\mathcal{A}}_2)$.
Step 3. Select $K^{\mathcal{A}}_j$, compute $M^{\mathcal{A}}_4 = h(x_j||T^{\mathcal{A}}_1||T^{\mathcal{A}}_2) \oplus K^{\mathcal{A}}_j$, $M^{\mathcal{A}}_5 = h(SID^{\mathcal{A}}_j||M^{\mathcal{A}}_4||T^{\mathcal{A}}_1||T^{\mathcal{A}}_2||K^{\mathcal{A}}_j)$.
Step 4. Send GWN $\{M_1, M_2, M_3, T_1, T^{\mathcal{A}}_2, ESID^{\mathcal{A}}_j, M^{\mathcal{A}}_4, M^{\mathcal{A}}_5\}$.
Step 5. GWN respondes $\{M_6, M_7, M_8, M_9, T_3\}$ to SN_j (i.e., the adversary), $\mathcal{A}$ computes $K^{\mathcal{A}}_i = M_7 \oplus h(x_j||T_3)$, $SK^{\mathcal{A}} = (K^{\mathcal{A}}_i \oplus K^{\mathcal{A}}_j)$, $M^{\mathcal{A}}_{10} = h(SK^{\mathcal{A}}||M_6||M_8||T_3||T^{\mathcal{A}}_4)$, sends $\{M_6, M_8, M^{\mathcal{A}}_{10}, T_3, T^{\mathcal{A}}_4\}$ to U_i.
Step 6. U_i will authenticate $\mathcal{A}$ successfully, that is, U_i thinks that she shares the session key with SN_m, while actually with SN_j (i.e., the adversary).

- *Time Complexity*: close to a legitimate sensor node.
- *The Scale of the Attack*: (a)→(j)→(g) as shown in Fig. 4. In the beginning, $\mathcal{A}$ compromises the sensor node SN_1 (marked in red circle as shown in (a) of Fig. 4), then exploits messages among U_1, SN_2 and GWN to impersonate SN_2 as above. Similarly, $\mathcal{A}$ can impersonate SN_2 to all users as shown in (j) of Fig. 4. When the number of the attacks is large enough, $\mathcal{A}$ can impersonate any sensor nodes to any users, as shown in (g) of Fig. 4.

4.9 Node Capture Attack Type IX

In Type IX, the adversary $\mathcal{A}$ exploits the insecure transmission or distribution of GWN's long term secret key to obtain the secret key. In this section, we show the details of Type IX via analysis of Das *et al.*'s scheme [49]:

- *Adversary's Capability*:
  - (1) "C6". Get SN_j's private key MK_{CH_j}.
  - (2) "C7". Register as a legitimate user U_i.
- *Attack Target*: the gateway.
- *Attack Consequence*: get GWN's long-term secret key.
- *Attack Steps*: note that $K_j = E_{MK_{CH_j}}(ID_i||ID_{CH_j}||X_s)$ is a special parameter related to GWN and SN_j and stored in U_i's smart card, where X_s is GWN's long-term secret key and MK_{CH_j} is SN_j's private key. Once $\mathcal{A}$ collude with U_i or register as a legitimate user U_i and get K_j from U_i's smart card, then $\mathcal{A}$ can obtain X_s via decrypting K_j with MK_{CH_j}.
- *Time Complexity*: $\mathcal{O}(T_S)$, where T_S is the operation time for symmetric encryption and decryption.
- *The Scale of the Attack*: (a)→(h) as shown in Fig. 4. In the beginning, $\mathcal{A}$ compromises the sensor node SN_1 (marked in red circle as shown in (a) of Fig. 4), then exploits U_1's smart card to get GWN's long-term secret key as above. After the first attack, $\mathcal{A}$ can obtain all unique secret parameters of U and SN, because their secret parameters are computed via this secret key. Thus, all participants and sessions are affected, as shown in (h) of Fig. 4.

4.10 Node Capture Attack Type X

In Type X, the adversary $\mathcal{A}$ with X_{SN_j} can modify session key between U and SN_j (or SN_m) without being noticed by any participants. After the attack, U and SN_j (or SNm) do not share the same session key and $\mathcal{A}$ can compute the same session key as U. But the authentication is finished successfully. Generally, this attack includes two situations:

(1) $\mathcal{A}$ can modify session keys between U and compromised sensor node $\mathsf{SN_j}$ due to users' ineffective verification to $\mathsf{CP_{SK/S}}$. Taking Amin *et al.'s* scheme [50] as an example, the last message of this scheme can be modified by $\mathcal{A}$ as $\{\mathsf{M}_8^{\mathcal{A}}, \mathsf{M}_9^{\mathcal{A}}, \mathsf{M}_{10}^{\mathcal{A}}, \mathsf{M}_{11}^{\mathcal{A}}\}$, where $\mathsf{M}_8^{\mathcal{A}} = \mathsf{h}(\mathsf{R}_2^{\mathcal{A}}) \oplus \mathsf{R}_3^{\mathcal{A}}$, $\mathsf{M}_9^{\mathcal{A}} = \mathsf{M}_9 \oplus \mathsf{R}_2 \oplus \mathsf{R}_2^{\mathcal{A}}$, $\mathsf{SK}^{\mathcal{A}} = \mathsf{h}(\mathsf{M}_6'|| R_2^{\mathcal{A}}||R_3^{\mathcal{A}})$, $\mathsf{M}_{10}^{\mathcal{A}} = \mathsf{h}(\mathsf{ID_i}||\mathsf{SK}^{\mathcal{A}}||\mathsf{R}_3^{\mathcal{A}})$, $\mathsf{M}_{11}^{\mathcal{A}} = \mathsf{M}_{11} \oplus \mathsf{h}(\mathsf{R}_2 \oplus R_3) \oplus h(R_2^{\mathcal{A}} \oplus \mathsf{R}_3^{\mathcal{A}})$, note that $\mathsf{ID_i}$ can be view as a known value to $\mathcal{A}$, $\mathsf{R}_2 = \mathsf{M}_5 \oplus \mathsf{h}(\mathsf{SK_{GWSN_j}})$ ($\mathsf{SK_{GWSN_j}}$ is $\mathsf{SN_j}$'s private key) and $\mathsf{R}_3 = \mathsf{M}_8 \oplus \mathsf{R}_2$. In this way, the authentication is finished successfully, yet $\mathsf{U_i}$ and $\mathsf{SN_j}$ do not share the same session key.

The scale of the attack is (a)→(c) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ compromises sensor node $\mathsf{SN_1}$ (marked in red circle as shown in (a)), and makes $\mathsf{SN_1}$ and $\mathsf{U_1}$ share different session keys as above. Similarly, $\mathcal{A}$ then can modify session keys between $\mathsf{SN_1}$ and $\mathsf{U_2}$. When the number of the attacks is large enough, $\mathcal{A}$ can modify session keys between all users and $\mathsf{SN_1}$ as shown in (c).

(2) In addition to problems in the first case, if $\mathsf{CP_{SK/U}}$ is transmitted insecurely too, the second situation occurs where $\mathcal{A}$ can modify all session keys between U and SN to make participants cannot share the same session keys, though the authentication is finished successfully. We use Amin *et al.*'s scheme [38] to show the attack:

- *Adversary's Capability*:
  (1) "C1". Eavesdrop SS_j and T_2 from message between $\mathsf{SN_j}$ and GWN, intercept and send message to GWN.
  (2) "C6". Get $\mathsf{SN_j}$'s secret key f_j.
- *Attack Consequence*: modify session key without noticed ($\mathsf{U_i}$ and $\mathsf{SN_m} (\mathsf{m} \neq \mathsf{j})$ share a different session key), meanwhile the authentication is finished successfully, and $\mathcal{A}$ can compute the same session key as $\mathsf{U_i}$.
- *Attack Steps*:

Step 1. Compute $\mathsf{h}(\mathsf{ID_i}) = \mathsf{SS_j} \oplus \mathsf{h}(\mathsf{f_j}||\mathsf{T_2})$.
Step 2. Eavesdrop $\mathsf{V_m}$ from GWN to $\mathsf{SN_m} (\mathsf{m} \neq \mathsf{j})$, compute $\mathsf{K_i} = \mathsf{V_m} \oplus \mathsf{h}(\mathsf{ID_i})$.
Step 3. Intercept $\{\mathsf{M_1}, \mathsf{K_{im}}, \mathsf{T_4}\}$ from GWN to $\mathsf{U_i}$.
Step 4. Generate $\mathsf{K}_{\mathsf{m}}^{\mathcal{A}}$ which has the same length as $\mathsf{K_m}$.
Step 5. Compute $\mathsf{SK}_{\mathsf{m}}^{\mathcal{A}} = \mathsf{h}(\mathsf{h}(\mathsf{ID_i})||\mathsf{SID_m}||\mathsf{K_i}||\mathsf{K}_{\mathsf{m}}^{\mathcal{A}})$.
Step 6. Compute $\mathsf{K}_{\mathsf{im}}^{\mathcal{A}} = \mathsf{K_i} \oplus \mathsf{K}_{\mathsf{m}}^{\mathcal{A}}$.
Step 7. Compute $\mathsf{M}_1^{\mathcal{A}} = \mathsf{h}(\mathsf{SK}_{\mathsf{m}}^{\mathcal{A}}||\mathsf{K}_{\mathsf{m}}^{\mathcal{A}}||\mathsf{T}_4^{\mathcal{A}})$.
Step 8. Send $\{\mathsf{M}_1^{\mathcal{A}}, \mathsf{K}_{\mathsf{im}}^{\mathcal{A}}, \mathsf{T}_4^{\mathcal{A}}\}$ to $\mathsf{U_i}$, then $\mathsf{U_i}$ will authenticate the message successfully.
Step 9. Intercept $\{\mathsf{M_2}\}$ from $\mathsf{U_i}$ to GWN.
Step 10. Compute $\mathsf{K_m} = \mathsf{K_{im}} \oplus \mathsf{K_i}$.
Step 11. Compute $\mathsf{M}_2^{\mathcal{A}} = \mathsf{M_2} \oplus \mathsf{h}(\mathsf{h}(\mathsf{h}(\mathsf{ID_i})||\mathsf{SID_m}||\mathsf{K_i}||\mathsf{K_m}) ||\mathsf{K_i}) \oplus \mathsf{h}(\mathsf{SK}_{\mathsf{m}}^{\mathcal{A}}||\mathsf{K_i})$.
Step 12. Send $\{\mathsf{M}_2^{\mathcal{A}}\}$ to GWN. Finally, $\mathsf{U_i}$ and $\mathsf{SN_m}$ do not share the same session key, yet $\mathsf{U_i}$ and $\mathcal{A}$ do.

- *Time Complexity*: $\mathcal{O}(6\mathsf{T_H})$.

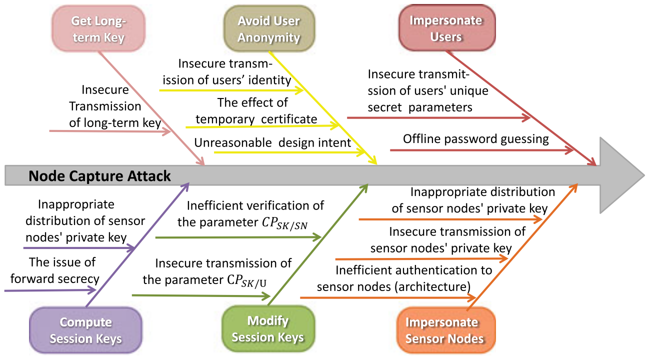


Fig. 6. The fishbone of node capture attacks.

- *The Scale of the Attack*: (a)→(b)→(e)→(f) as shown in Fig. 4. In the beginning, the adversary $\mathcal{A}$ compromises $\mathsf{SN_1}$ (marked in red circle as shown in (a) of Fig. 4) and gets some parameters of U_1 to modify session key between U_1 and $\mathsf{SN_1}$ as above as shown in (b) of Fig. 4. Similarly, $\mathcal{A}$ can get $\mathsf{U_2}$'s $h(ID_2)$ and modify session key between U_2 and $\mathsf{SN_1}$. With the increase in number of the attacks, $\mathcal{A}$ can get all users' useful parameters and modify session keys between all users and $\mathsf{SN_1}$. When the number of the attacks is large enough, $\mathcal{A}$ finally can modify session keys between all users and all sensor nodes as shown in (f) of Fig. 4.

Note that, in the first situation, the adversary $\mathcal{A}$ can only modify session keys between compromised sensor node $\mathsf{SN_j}$ and all users. In the second situation, $\mathcal{A}$ finally can modify session keys between all sensor nodes and all users.

5 Suggestions to Node Capture Attacks

Sensor nodes are usually deployed in unattended environments, thus it is easy for an adversary to breach some sensor nodes and extract the data stored in them. Based on this reality, it is very important to ensure the security of the system after the sensor node is compromised. Much effort has been taken to design a secure scheme resisting such an attack, while most attempts failed. In this section, we summarize the rationales for node capture attacks, put forward some suggestions to avoid such attacks.

From attack consequences of Table 4, the adversary can 1) compute session keys from Type I and II, 2) impersonate users from of Type III and IV, 3) avoid user anonymity from Type V, 4) impersonate sensor nodes from Type VI, VII and VIII, 5) get the long-term secret key from Type IX, 6) modify session keys from Type X. Based on these six consequences, we figure out their causes from the vulnerabilities exploited of Table 4. With the six consequences and their causes, we draw the fishbone of node capture attacks, as shown in Fig. 6. It clarifies the causes of node capture attacks. Focusing on these listed causes in Fig. 6, these five causes, namely insecure transmission of long-term secret key, insecure transmission of users' identity, insecure transmission of $\mathsf{FUSP_{G/U}}$, insecure transmission of $\mathsf{CP_{SK/SN}}$, and insecure transmission of SN's private key, can be concluded as insecure transmission of $\mathsf{FUSP_{G/U}}$ or some parameters. Then these listed causes of node capture attacks in Fig. 6 can be summarized to eight aspects:

unreasonable design intent, insecure communication architecture, inappropriate distribution of SN's private key, insecure transmission of $FUSP_{G/U}$ or some parameters, inefficient verification of SN or $CP_{SK/SN}$, the issue of forward secrecy, the issue of offline dictionary attacks and the issue of temporary certificate. Suggestions to these eight issues are as follows:

- *Unreasonable Design Intent.* Note that user identity, unique secret parameter ($FUSP_{G/U}$ and $TUSP_{G/U}$) cannot be known to sensor nodes, so do not let the gateway send these parameters to sensor nodes.
- *Insecure Communication Architecture.* From the viewpoint of node capture attacks, model (b) and model (c) of Fig. 5 are insecure: model (b) is likely to result in node capture attack Type VII as shown in Section 4.7. Model (c) is likely to result in node capture attack Type VIII as shown in Section 4.8. Furthermore, both model (b) and model (c) is bound to led to node capture attack Type X. Following Wang *et al.*'s research [6], model (a) of Fig. 5 is recommended.
- *Inappropriate Distribution of* SN*'s Private Key.* The distribution of SN's private key is a basic factor to the security of the system. Looking back Fan *et al.*'s scheme [29], it is very dangerous to have all sensor nodes share a same secret key. We recommend that let $h(ID_{SN_j}||x)$ be SN's private key, this method has been accepted by most schemes [28], [30], [34], [54], [55]. In some of schemes, such as Gope *et al.*'s [56], GWN assigns a random unique secret number to SN as their private key. In this way, GWN must store the parameters related to the private keys of SN, which consumes more resources. Thus this method is not recommended.

  Following this principle, the private key of sensor nodes in Fan *et al.*'s scheme [29] should be $h(ID_{SN_j}||x)$ rather than a common shared key.
- *Insecure Transmission of Some Parameters,* including the private key X_{SN_j} of SN_j, unique secret parameters/identity ID of users, and long-term secret key.

  The ways that the unique secret parameters are transmitted are varied from one protocol design to another. It is difficult to generalize, some basic principles are as follows:

  - Transmitting these parameters with "XOR" or symmetric encryption operation is dangerous, see Sections 4.3, 4.5, 4.6 and 4.9. We recommend to protect these parameters (denoted as $Impor_{Par}$) in a form of $h(Impor_{Par}||*)$, where $*$ denotes any parameters. Particularly, when $Impor_{Par}$ is $FUSP_{G/U}$, "$*$" has to include $TUSP_{G/U}$ (it is constructed by a public-key technique).

    Following this principle, we can fix Li *et al.*'s scheme [34] by setting $M_5 = h(M_1||M_3) \oplus r_i$, $M_{14} = h(M_3||M_1) \oplus r_g$ and all ID_i in the parameters that GWN sends to SN_j be replaced with $h(ID_i||K_{GWN-S})$, where M_1 is $FUSP_{G/U}$ and M_5 is $TUSP_{G/U}$. In this way, the adversary cannot follow the steps in Section 4.3 to extract $FUSP_{G/U}$, so this scheme can resist against the attack Type III, IV, V and X.
  - In some occasions, ID and X_{SN_j} need to be transmitted with the operation "XOR", we recommend to transmit them in the form of $ID \oplus h(TUSP_{G/U}||*)$ and $X_{SN_j} \oplus h(TUSP_{G/SN}\ ||*)$ (or $X_{SN_j} \oplus TUSP_{G/SN} \oplus *$), respectively. Note that $TUSP_{G/U}$ is constructed by a public-key technique.

    Following this principle, we can fix Li *et al.*'s scheme [57] by setting $DID_U = ID_U \oplus h(D_2||D_1)$, $D_3 = SN_{id} \oplus h(B_2||D_2)$, where B_2 is $FUSP_{G/U}$ and D_2 is $TUSP_{G/U}$. In this way, the adversary cannot extract $FUSP_{G/U}$, so this improved scheme is secure against the attack Type III and IV.
- *Inefficient Verification of* SN *or* $CP_{SK/SN}$. It contains two aspects: 1) GWN fails to authenticate SN. 2) users fail to verify $CP_{SK/SN}$. To the first aspect, the first thing is to use a proper communication model. Then, do not merely rely on X_{SN_j} to finish the authentication between SN_j and GWN, a temporary challenge, such as R_{GS} in Fig. 3, is necessary. Specifically, let $Auth_3$ at least contain $FUSP_{G/SN}$, $TUSP_{G/SN}$ and $CP_{SK/SN}$. It can stop the adversary from forging messages to fool GWN. To the second aspect, a public-key technique is required to construct $TUSP_{G/U}$, and $Auth_4$ should at least contain $TUSP_{G/U}$ and $CP_{SK/SN}$. This method can stop the adversary from modifying session keys (see Section 4.10).

  Following above principle, to improve Amin *et al.'s* scheme [38], we first introduce ECC-based public-key technique with a pair of private/public key (X_{GWN}, $X_{GWN} \cdot P = Y$), where P is a point on an elliptic curve which is built over prime finite field F_p. Next, we construct $TUSP_{G/U}$ as $TM_2 = K_i \cdot Y$, and let U_i send $TM_1 = K_i \cdot P$ to GWN. Till now, U_i and GWN share the $TUSP_{G/U}$ (GWN can obtain TM_2 by computing $X_{GWN} \cdot TM_1$). Next, we set the $Auth_3$ of Amin *et al.*' s scheme [38] W_j be $h(f_j||TM_2||K_j)$, where f_j is $FUSP(G/SN)$, K_j is $TUSP(G/SN)$ and TM_2 is $CP(SK/SN)$. Finally, we set the $Auth_4$, i.e., M_1, be $h(TM_2||K_j||K_g||T_4)$.
- *Forward Secrecy.* To achieve forward secrecy, at least two modular multiplication or point multiplication operations are needed on sensor node [12]. Once a scheme achieves forward secrecy, then it is resistant against the attack Type I and Type II.

  Following this principle, we continue to improve Aim *et al.*'s scheme [38]. We let SN_j compute two point multiplication operations: $TM_3 = K_j \cdot P$ and $TM = K_i \cdot TM_1$, and set session keys $SK = h(TM_1||TM_3||TM)$. Note that TM is not transmitted in any channel. In this way, Aim *et al.*'s scheme [38] can achieve forward secrecy and resist against the attack Type I and Type II.
- *Offline Dictionary Attacks.* To resist offline dictionary attacks, the public-key algorithm is indispensable [52]. Yet, there is a subtlety worth noting: when accessing offline dictionary attacks, in addition to focusing on VP in login request initiated by U, special attention should be taken to $Auth_4$ and parameters with $FUSP_{G/U}$ in the channel, and this is often ignored by the protocol designer. As we have shown in Section 4.4, the adversary can exploit $Auth_4$ (M_8)

to carry out a dictionary attack successfully in the Jiang *et al.*'s scheme [30]. A recommended solution is to use $\mathsf{FUSP}_{G/U}$ in a form of $\mathsf{h}(\mathsf{FUSP}_{G/U}||\mathsf{TUSP}_{G/U}||*)$, where $*$ denotes any valid numbers, and $\mathsf{TUSP}_{G/U}$ is constructed by a public-key technique.

Following this principle, we can improve Jiang *et al.*'s scheme [30]. First, we need to construct a $\mathsf{TUSP}_{G/U}$. The way to construct such a $\mathsf{TUSP}_{G/U}$ has been introduced above, therefore, the details are omitted. Second, we let $\mathsf{M}_8 = \mathsf{h}(\mathsf{d}_i||\mathsf{TUSP}_{G/U}||\mathsf{ID}_i||\mathsf{K}_j)$. In this way, Jiang *et al.*'s scheme [30] is resistant to the attack Type V.

– *The Issue of Temporary Certificate.* Applying temporary certificate algorithm to multi-factor user authentication schemes leads many problems [52], how to avoid these problems is still an open question. A simple way is that do not use temporary certificate technique. Actually, many schemes that do not use temporary certificate technique [26], [27], [28], [57] achieve as least the same security as those using this technique [37], [50], [56].

To design a secure authentication scheme that is resistant to node capture attacks, the above eight challenges should be taken into account. The specific method to achieve the above suggestions may be different from scheme to scheme, but we summarize some common principles against node capture attacks: 1) Regarding "Insecure communication architecture", model (b) and model (c) of Fig. 5 are not secure against node capture attacks, and model (a) is recommended. 2) Users' identity and unique secret parameter should be kept anonymous to sensor nodes. 3) It is recommended to set $\mathsf{h}(\mathsf{ID}_{SN_j}||\mathsf{x})$ as SN's private key. Furthermore, as shown in Appendix A , which can be found at https://bit.ly/2VjHqY1 and also on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TDSC.2020.2974220, we take Li *et al.*'s scheme [57] as an example to show a viable way to follow the above principles to avoid node capture attacks.

6 A Comparative Evaluation of Exiting Schemes for WSNs

Based on our taxonomy of node capture attacks in Section 3, we naturally improve our evaluation criteria by expanding the criterion "resistance to node capture attacks" into ten sub-criteria. We then perform a large-scale assessment of 61 multi-factor user authentication schemes for WSNs under our expanded criteria set and our attack model in Table 5. The selected schemes usually represent a typical attack or have attracted much attention and lead many new enhanced versions. This comparison gives a fair and comprehensive evaluation of existing schemes. Unsurprisingly, two early schemes, which were proposed around the year of 2005 when Benenson *et al.* [18] for the first time introduce node capture attacks into remote user authentication, are worse than other schemes. As time goes by, the situation gets better, which is in line with our understanding on the development of things. From Table 5, it is easy to see that so far no scheme meets all evaluation criteria after nearly ten years of research. The scheme with the best performance proposed by Li *et al.* [58] can only achieve at most 20 criteria, highlighting the unsatisfactory situation of user authentication schemes for WSNs.

When dividing the evaluation criteria into two parts, i.e., ideal attributes and security requirements, we can see another trend in the development of user authentication schemes, that is, the schemes' performance in ideal attributes gets better and better. The challenge in satisfying the requirements of ideal attributes is to design schemes without relying on clock synchronization. Compared with the realization of ideal attributes, meeting the security requirements is more difficult. Every criteria of security requirements except S1 are all difficult to meet. In most cases, offline dictionary attacks are the main kinds of attacks for S5, and it can be stopped by applying public-key techniques correctly [12], [52]. In WSNs, S4 "resistance to known attacks" is becoming difficult as the complexity of system increases. According to Li *et al.*'s recent study [80], the administrator of the gateway can exploit the user's login request as a verifier to guess victims' passwords. Most schemes are vulnerable to such an attack and thus cannot satisfy the criterion S2 "No Password Exposure". S3 "forward secrecy" is a tricky problem in WSNs because of the recourse-limited sensor nodes. "How to efficiently achieve forward secrecy in user authentication scheme for WSNs" is still an open issue.

Among all criteria, S6 "resistance to node capture attacks" is the hardest criterion to be achieved. As shown in Table 5, only two schemes meet S6. However, this trend cannot be reflected well under other evaluation criteria sets. For example, the schemes of Li *et al.* (2018 JNCA) [34], Wu *et al.* (2017 PPNA) [37], Wang *et al.* (2018 Sensors) [28], Jiang *et al.* (2017 IEEE Access) [30] and Das *et al.* (2016 SCN) [65] are thought of being resistant to node capture attacks under Wang *et al.*'s criteria set [6] where node capture attacks are included in the criterion "resistance to known attacks", but these schemes are demonstrated that they cannot resist against node capture attacks under our criteria set. All this highlights the urgency and significance of understanding the failure in node capture attacks and the difficulty in designing a user authentication scheme for WSNs resistant against node capture attacks. Furthermore, each sub-criterion of S6 is met or unmet by at least ten schemes. This indicates that each of the ten sub-criteria is necessary and our taxonomy of node capture attacks is reasonable. We present a detailed discussion on S6 in Appendix B, available at https://bit.ly/2VjHqY1.

7 Conclusion

In this paper, we have taken the first substantial step towards systematically exploring node capture attacks against user authentication protocols for WSNs. We first define the adversary model, and then develop a detailed and through evaluation criteria including the effects of node capture attacks. We then categorize node capture attacks into ten different types in terms of the attack targets, adversary's capabilities and vulnerabilities exploited. Next, we elaborate on each type of attacks through examining 11 typical vulnerable protocols and investigate the conrresponding countermeasures. Finally, we extend our evaluation criteria and conduct a large-scale comparative measurement of 61 representative user authentication schemes for WSNs. Among those schemes, only two are secure against node capture attacks, highlighting the difficulty in designing node-capture-attack resistant user authentication schemes for WSNs and demonstrating the significance of our systematic study on node capture attacks.

TABLE 5
Security and Efficiency Comparison Among Relevant User Authentication Schemes

Protocol	Year	Ref.	Evaluation Criteria																					Computational Cost		
			D1	D2	D3	D4	D5	D6	S1	S2	S3	S4	S5	S6										User	Gateway	Sensor Node
														I	II	III	IV	V	VI	VII	VIII	IX	X			
Li et al.	2019	[58]	√	√	√	√	√	√	√	×	√	√	√	√	√	√	√	√	√	√	√	√	√	$3T_P + 10T_H + T_B$	$T_P + 8T_H$	$2T_P + 4T_H$
Gupta et al.	2019	[59]	√	√	√	×	√	√	×	×	×	×	×	×	√	√	√	√	√	√	√	√	×	$7T_H$	$5T_H$	$3T_H$
Li et al.	2018	[57]	√	√	√	√	√	√	√	×	√	×	√	√	√	×	×	√	√	√	√	√	×	$3T_P + 7T_H + T_B$	$T_P + 6T_H$	$2T_P + 4T_H$
Srinivas et al.	2018	[11]	√	√	√	×	√	×	√	×	√	×	×	√	√	√	×	×	√	√	√	√	×	$2T_C + 14T_H + T_B$	$10T_H$	$2T_C + 6T_H$
Srinivas et al.	2018	[60]	-	√	√	×	√	√	×	×	√	√	√	√	√	√	-	√	√	√	√	√	×	$2T_P + T_B + 6T_H$	$6T_H + T_S$	$2T_P + T_S + 4T_H$
Wazid et al.	2018	[5]	√	√	√	×	√	×	√	√	×	√	×	×	√	√	×	×	√	√	√	√	√	$T_B + 13T_H + 2T_S$	$5T_H + 4T_S$	$4T_H + 2T_S$
Wu et al.	2018	[54]	×	√	√	√	√	√	√	×	×	×	×	×	√	×	×	√	√	√	√	√	√	$11T_H$	$15T_H$	$5T_H$
Amin et al.	2018	[50]	√	×	√	×	√	√	√	×	×	×	×	×	√	√	×	√	√	√	√	√	×	$12T_H$	$15T_H$	$6T_H$
Amin et al.	2018	[38]	√	√	√	×	√	√	×	√	×	√	×	×	√	√	×	×	√	√	√	√	×	$T_B + 14T_H$	$16T_H$	$4T_H$
Li et al.	2018	[34]	√	√	√	√	√	√	√	×	×	√	×	×	√	×	×	×	√	√	√	√	×	$2T_P + T_B + 7T_H$	$T_P + 9T_H$	$4T_H$
Ali et al.	2018	[39]	√	√	√	×	√	√	×	×	×	×	×	×	√	√	×	×	√	√	√	√	×	$T_B + 7T_H + 2T_S$	$11T_H + 5T_S$	$4T_H + T_S$
Challa et al.	2018	[40]	√	√	√	×	√	√	√	√	×	×	×	×	√	√	√	×	√	×	√	√	×	$T_P + T_B + 9T_H$	$T_P + 4T_H$	$5T_H$
Xiong et al.	2017	[61]	√	√	√	×	√	×	√	×	√	×	×	√	√	√	×	×	√	√	√	√	√	$9T_H + 2T_S$	$11T_H + 2T_S$	$5T_H + T_S$
Wu et al.	2017	[62]	×	√	√	√	√	√	√	×	√	√	×	√	√	√	×	√	√	√	√	√	√	$2T_P + T_S + 12T_H$	$2T_S + 11T_H$	$2T_P + T_S + 4T_H$
Wang et al.	2017	[28]	√	√	√	×	√	√	√	×	√	√	√	√	√	√	×	√	√	√	√	√	√	$3T_P + T_B + 10T_H$	$T_P + 11T_H$	$2T_P + 4T_H$
Srinivas et al.*	2017	[27]	√	√	√	×	√	√	×	×	×	√	×	×	√	√	√	√	√	√	√	√	×	$T_B + 9T_H$	$13T_H$	$6T_H$
Wazid et al.	2017	[1]	√	√	√	×	√	×	√	×	×	×	×	×	√	×	×	×	√	√	√	√	×	$T_B + 9T_H + T_S$	$11T_H + 2T_S$	$7T_H + T_S$
Jung et al.	2017	[33]	√	√	√	×	√	√	×	×	×	×	×	×	×	×	×	×	×	√	√	×	×	$T_B + 7T_H$	$9T_H$	$4T_H$
Moon et al.	2017	[63]	-	√	√	×	√	√	√	×	√	×	√	√	√	×	-	×	√	√	√	√	×	$3T_P + T_B + 6T_H$	$T_P + 7T_H + T_S$	$2T_P + 4T_H + T_S$
Wu et al.*	2017	[37]	×	√	√	×	√	√	√	×	×	×	×	×	√	×	×	×	×	√	√	√	×	$8T_H$	$10T_H$	$3T_H$
Tai et al.	2017	[47]	√	√	√	×	√	√	√	×	×	√	×	×	√	√	√	√	√	√	×	√	×	$8T_H$	$10T_H$	$6T_H$
Dhillon et al.	2017	[41]	√	√	√	×	√	√	×	×	×	×	×	×	√	×	×	×	×	√	×	√	×	$T_B + 9T_H$	$6T_H$	$7T_H$
Jiang et al.	2017	[30]	√	√	√	×	√	√	√	√	×	√	√	×	√	√	×	×	√	√	√	√	√	$T_E + T_B + 8T_H$	$T_E + 12T_H$	$5T_H$
Park et al.	2016	[26]	√	√	√	×	√	√	×	×	√	√	×	√	√	√	×	√	√	√	√	√	√	$2T_P + T_B + 10T_H$	$11T_H$	$2T_P + 4T_H$
Gope et al.	2016	[56]	√	√	√	×	√	√	×	√	√	×	×	√	√	√	√	√	√	√	√	√	√	$12T_H$	$9T_H$	$3T_H$
Reddy et al.	2016	[64]	×	√	√	√	√	√	√	×	√	√	√	√	√	√	√	√	√	√	√	√	×	$3T_P + 10T_H$	$2T_P + 7T_H$	$2T_P + 5T_H$
Chang et al.†	2016	[55]	×	√	√	×	√	√	×	×	√	√	×	√	√	√	√	√	√	√	√	√	×	$2T_P + 5T_H$	$8T_H$	$2T_P + 5T_H$
Das et al.	2016	[65]	√	√	√	×	√	√	×	×	√	√	×	√	√	√	×	√	√	√	√	√	×	$2T_P + T_B + 12T_H$	$2T_P + 9T_H$	$10T_H$
Lu et al.	2016	[66]	√	√	√	×	√	√	√	×	×	√	×	×	√	√	√	×	×	√	√	√	√	$7T_H + 2T_S$	$9T_H + 4T_S$	$3T_H + 2T_S$
Choi et al.	2016	[67]	-	×	√	×	√	√	×	√	√	×	×	√	√	√	-	√	√	√	√	√	×	$2T_P + T_B + 7T_H + T_S$	$8T_H + 2T_S$	$2T_P + 5T_H + T_S$
Park et al.	2016	[68]	-	×	√	×	√	√	×	√	√	×	√	√	√	√	-	√	√	√	√	√	×	$2T_P + T_B + 6T_H$	$6T_H + 2T_S$	$2T_P + 6T_H + T_S$
Kumari et al.	2016	[42]	×	√	√	×	√	√	×	×	√	√	×	√	√	√	√	√	√	×	√	√	×	$2T_P + 4T_H + 2T_S$	$6T_H + T_S$	$2T_P + 3T_H$
Jiang et al.	2016	[69]	×	√	√	×	√	√	×	×	×	√	√	×	√	√	√	×	√	√	√	√	×	$2T_P + 8T_H$	$T_P + 9T_H$	$6T_H$
Amin et al.	2016	[70]	√	√	√	×	√	√	×	√	×	×	×	×	√	√	×	×	√	√	√	√	√	$12T_H$	$15T_H$	$5T_H$
Amin et al.*	2016	[71]	√	√	√	×	√	√	×	×	×	×	×	×	√	×	×	×	√	√	√	√	√	$7T_H$	$8T_H$	$5T_H$
Farash et al.	2016	[45]	√	√	√	×	√	√	×	×	×	×	×	×	√	√	×	√	√	√	×	√	×	$11T_H$	$14T_H$	$7T_H$
Kumari et al.	2016	[2]	√	√	√	√	×	√	×	×	×	√	×	×	√	√	√	√	√	√	√	√	×	$9T_H$	$8T_H$	$6T_H$
He et al.	2015	[72]	×	×	√	×	√	√	×	√	×	√	×	×	√	√	√	×	√	√	√	√	×	$8T_H$	$9T_H$	$6T_H$
Das et al.	2015	[43]	√	√	√	×	√	√	×	×	×	×	×	×	√	×	√	×	√	×	√	√	×	$8T_H + T_S$	$2T_H + T_S$	$2T_H + T_S$
Chang et al.	2015	[25]	√	√	√	×	√	√	√	×	×	×	×	×	√	√	×	√	√	√	√	√	√	$11T_H$	$10T_H$	$4T_H$
Jiang et al.	2015	[35]	×	√	√	×	√	√	√	×	×	×	×	×	√	√	×	√	√	√	√	√	√	$7T_H$	$10T_H$	$5T_H$
Choi et al.	2014	[36]	√	√	√	×	√	√	×	×	√	√	×	√	√	√	√	×	√	√	√	√	×	$3T_P + 9T_H$	$2T_P + 6T_H$	$T_P + 5T_H$
Kim et al.	2014	[24]	√	√	√	×	√	√	×	×	×	×	×	×	×	×	×	√	×	√	√	√	×	$8T_H$	$8T_H$	$2T_H$
Turkanovic et al.	2014	[73]	√	√	√	×	√	√	×	×	×	×	×	×	√	×	×	×	√	√	×	√	×	$7T_H$	$7T_H$	$5T_H$
Turkanovic et al.	2013	[74]	√	√	√	×	×	√	×	×	×	×	×	×	√	√	√	√	√	√	√	×	-	$4T_H + T_S$	$2T_H + 3T_S$	$T_H + T_S$
Yoon and Kim	2013	[75]	-	×	√	×	√	√	×	√	×	×	√	×	√	√	-	×	√	√	√	√	×	$4T_H + T_S$	$4T_H + 2T_S$	$3T_H + T_S$
Xue et el.	2013	[76]	√	√	√	×	√	√	×	×	×	×	×	×	√	√	×	×	√	√	√	√	×	$9T_H$	$14T_H$	$6T_H$
Shi et el.	2013	[53]	√	√	√	×	√	√	×	×	√	√	×	√	√	√	√	×	√	√	×	√	×	$3T_P + 6T_H$	$T_P + 5T_H$	$2T_P + 4T_H$
He et el.	2012	[31]	×	√	×	×	√	√	×	×	×	×	×	×	√	√	√	×	√	√	√	√	√	$3T_H + T_S$	$4T_H + 2T_S$	$2T_H + T_S$
Das et el.	2012	[49]	√	√	√	×	×	√	×	×	×	×	×	×	√	√	√	√	√	√	√	×	-	$5T_H + T_S$	$3T_H + T_S$	$2T_H + T_S$
Kumari et el.	2012	[44]	√	×	√	×	√	√	√	×	×	×	×	×	√	√	√	×	√	×	√	√	√	$3T_H + 2T_S$	$3T_H$	$T_H + 2T_S$
Vaidya et el.	2012	[23]	√	×	√	×	√	√	√	×	×	×	×	×	×	√	×	√	×	√	√	×	×	$7T_H$	$6T_H$	$2T_H$
Kumari et el.	2011	[77]	√	×	√	×	√	√	×	√	×	√	×	×	√	√	√	×	√	√	√	√	√	$6T_H + T_S$	$3T_H + 3T_S$	$2T_H + 3T_S$
Fan et el.	2011	[29]	√	×	√	×	×	√	×	×	×	×	×	×	×	√	√	√	×	√	√	√	√	$8T_H$	$6T_H$	$2T_H$
Yoon et el.	2011	[32]	-	×	×	×	√	√	×	√	×	×	×	×	×	√	-	×	×	×	√	√	×	$3T_H$	$4T_H$	$2T_H$
Yeh et el.	2011	[78]	√	×	√	×	×	×	×	√	×	×	×	×	×	×	√	×	×	×	√	√	×	$2T_P + 4T_H$	$3T_P + 4T_H$	$2T_P + 3T_H$
Vaidya et el.	2010	[19]	√	×	×	×	×	√	√	×	×	×	×	×	×	√	×	×	×	×	×	√	-	$6T_H$	$4T_H$	$2T_H$
He et el.	2010	[79]	√	×	×	×	×	√	×	√	×	×	×	×	×	×	√	×	×	×	×	√	-	$5T_H$	$5T_H$	T_H
Das	2009	[7]	√	×	×	×	×	√	×	×	×	×	×	×	×	√	√	×	×	×	×	√	-	$2T_H$	$4T_H$	T_H
Tseng et al.	2007	[21]	√	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	√	-	T_H	$3T_H$	T_H
Wong et al.	2006	[20]	√	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	√	-	0	$3T_H$	T_H

*: the case 1 of their scheme; †: the $\mathcal{P}_2$ of their scheme.
Some schemes do not describe dynamic sensor node addition or smart card revocation phases directly, but they do support the two phases, and thus meet D2. T_E, T_P, T_C, T_B, T_H, T_S denote the operation time for modular exponentiation, elliptic curve point multiplication, chebysev chaotic-map, fuzzy extracting biometric data, hash, and symmetric encryption, respectively, Some lightweight operations like XOR and $\|$ are omitted.
"√" denotes the scheme can provide the corresponding attribute. "×" denotes the scheme cannot provide the corresponding attribute. "-" means the attribute is not applied to the scheme. For example, if the scheme dose not create a session key after authentication, then it dose not make sense to discuss session key related security; if the scheme dose not use password, then it dose not make sense to discuss offline dictionary guessing attacks; if the message sent to users is an "acknowledgment" containing no sensitive parameters, then it dose not make sense to discuss whether the adversary can modify this message.

ACKNOWLEDGMENT

The authors thank Dr. Yanhong Xu and the anonymous reviewers for their invaluable comments. This research is supported by the National Key Research and Development Plan of China under Grant No. 2018YFB0803605, and by the National Natural Science Foundation of China under Grant No.61802006, and by the National Research Foundation, Prime Minister's Office, Singapore under its Strategic Capability Research Centres Funding Initiative and Singapore Ministry of Education under Research Grant MOE2016-T2-2-014(S)

REFERENCES

[1] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Trans. Depend. Secur. Comput.*, 2017. to be published, doi:10.1109/TDSC.2017.2764083.

[2] S. Kumari and H. Om, "Authentication protocol for wireless sensor networks applications like safety monitoring in coal mines," *Comput. Netw.*, vol. 104, no. C, pp. 137–154, 2016.

[3] J. Wei, X. Hu, and W. Liu, "An improved authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 6, pp. 3597–3604, 2012.

[4] X. Yang *et al.*, "A lightweight authentication scheme for vehicular ad hoc networks based on msr," *Veh. Commun.*, vol. 15, no. 16–27, 2019.

[5] M. Wazid, A. K. Das, M. K. Khan, A. D. Al-Ghaiheb, N. Kumar, and A. Vasilakos, "Design of secure user authenticated key management protocol for generic IoT networks," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 269–282, Feb. 2018.

[6] D. Wang, W. Li, and P. Wang, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Trans. Ind. Inf.*, vol. 14, no. 9, pp. 4081–4092, Sep. 2018.

[7] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.

[8] Y. Zhang, Y. Xiang, and X. Huang, "Password authenticated group key exchange: A cross-layer design," *ACM Trans. Internet Technol.*, vol. 15, no. 4, pp. 24:1–24:20, 2016.

[9] W. Meng, Y. Wang, D. S. Wong, S. Wen, and Y. Xiang, "Touch behavioral user authentication based on web browsing on smartphones," *J. Netw. Comput. Appl.*, vol. 117, pp. 1–9, 2018.

[10] E. Erdem and M. T. Sandıkkaya, "OTPaaS—one time password as a service," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 3, pp. 743–756, Mar. 2019.

[11] J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, "Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial Internet of Things," *IEEE Trans. Dependable Secure Comput.*, 2018. to be published, doi: 10.1109/TDSC.2018.2857811.

[12] C. Ma, D. Wang, and S. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *Int. J. Commun. Syst.*, vol. 27, no. 10, pp. 2215–2227, 2012.

[13] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 708–722, Jul./Aug. 2018.

[14] D. W. Carman, P. S. Kruus, and B. J. Matt, "Constraints and approaches for distributed sensor network security," Cryptographic Technologies Group, Trusted Information System, NAI Labs. DARPA Project report, 2000, vol. 1, no. 1. [Online]. Available: http://download.nai.com

[15] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. IEEE Symp. Secur. Privacy* 2003, pp. 197–213.

[16] L. Eschenauer and V. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. 9th ACM Conf. Comput. Commun. Secur.* 2002, pp. 41–47.

[17] W. Du, J. Deng, Y. Han, P. Varshney, J. Katz, and A. Khalili, "A pairwise key predistribution scheme for wireless sensor networks," *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 2, pp. 228–258, 2005.

[18] Z. Benenson, N. Gedicke, and O. Raivio, "Realizing robust user authentication in sensor networks," in *Proc. Real-World Wireless Sensor Netw.*, 2005, vol. 14, pp. 52–56.

[19] B. Vaidya, J. J. Rodrigues, and P. J. Hyuk, "User authentication schemes with pseudonymity for ubiquitous sensor network in ngn," *Int. J. Commun. Syst.*, vol. 23, no. 9–10, pp. 1201–1222, 2010.

[20] K. H. M. Wong, Y. Zheng, J. Cao, and S. Wang, "A dynamic user authentication scheme for wireless sensor networks," in *Proc. IEEE Int. Conf. Sensor Netw.*, 2006, pp. 244–251.

[21] H. R. Tseng, R. H. Jan, and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks," in *Proc. IEEE Global Telecommun. Conf.*, 2007, pp. 986–990.

[22] L. C. Ko, "A novel dynamic user authentication scheme for wireless sensor networks," in *Proc. IEEE Int. Symp. Wireless Commun. Syst.*, 2008, pp. 608–612.

[23] B. Vaidya, D. Makrakis, and H. Mouftah, "Two factor mutual authentication with key agreement in wireless sensor networks," *Secur. Commun. Netw.*, vol. 9, no. 2, pp. 171–183, 2016.

[24] J. Kim, D. Lee, W. Jeon, Y. Lee, and D. Won, "Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks," *Sensors*, vol. 14, no. 4, pp. 6443–6462, 2014.

[25] I. Chang, T. Lee, T. Lin, and C. Liu, "Enhanced two-factor authentication and key agreement using dynamic identities in wireless sensor networks," *Sensors*, vol. 15, no. 12, pp. 29 841–29 854, 2015.

[26] Y. Park and Y. Park, "Three-factor user authentication and key agreement using elliptic curve cryptosystem in wireless sensor networks," *Sensors*, vol. 16, no. 12, p. 2123, 2016.

[27] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," *Ad Hoc Netw.*, vol. 54, no. C, pp. 147–169, 2017.

[28] C. Wang, G. Xu, and J. Sun, "An enhanced three-factor user authentication scheme using elliptic curve cryptosystem for wireless sensor networks," *Sensors*, vol. 17, no. 12, p. 2946, 2017.

[29] R. Fan, D. He, X. Pan, and L. Ping, "An efficient and dos-resistant user authentication scheme for two-tiered wireless sensor networks," *J. Zhejinag Univ. Sci. C*, vol. 12, no. 7, pp. 550–560, 2011.

[30] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," *IEEE Access*, vol. 5, pp. 3376–3392, 2017.

[31] D. He, "Robust biometric-based user authentication scheme for wireless sensor networks," *Ad Hoc Sensor Wireless Netw.*, vol. 25, no. 3, pp. 309–321, 2012.

[32] E.-J. Yoon and K.-Y. Yoo, "A new biometric-based user authentication scheme without using password for wireless sensor networks," in *Proc. IEEE Int. Workshops Enabling Technologies: Infrastructure Collaborative Enterprises*, 2011, pp. 279–284.

[33] J. Jung, J. Moon, D. Lee, and D. Won, "Efficient and security enhanced anonymous authentication with key agreement scheme in wireless sensor networks," *Sensors*, vol. 17, no. 3, pp. 644–665, 2017.

[34] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K. K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things environments," *J. Netw. Comput. Appl.*, vol. 103, pp. 194–204, 2018.

[35] Q. Jiang, J. Ma, X. Lu, and Y. Tian, "An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks," *Peer Peer Netw. Appl.*, vol. 8, no. 6, pp. 1070–1081, 2015.

[36] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 14, no. 6, pp. 10081–10106, 2014.

[37] F. Wu *et al.*, "An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment," *J. Netw. Comput. Appl.*, vol. 89, pp. 72–85, 2017.

[38] R. Amin, S. K. H. Islam, N. Kumar, and K. K. R. Choo, "An untraceable and anonymous password authentication protocol for heterogeneous wireless sensor networks," *J Netw. Comput. Appl.*, vol. 104, pp. 133–144, 2018.

[39] R. Ali, A. K. Pal, S. Kumari, M. Karuppiah, and M. Conti, "A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring," *Future Gener. Comput. Syst.*, vol. 84, pp. 200–215, 2018.

[40] S. Challa *et al.*, "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks," *Comput. Elect. Eng.*, vol. 69, pp. 534–554, 2018.

[41] P. K. Dhillon and S. Kalra, "Secure multi-factor remote user authentication scheme for Internet of Things environments," *Int. J. Commun. Syst.*, vol. 30, no. 16, p. e3323, 2017.

[42] 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 Gener. Comput. Syst.*, vol. 63, pp. 56–75, 2016.

[43] A. K. Das, "A secure and efficient user anonymity-preserving three-factor authentication protocol for large-scale distributed wireless sensor networks," *Wireless Pers. Commun.*, vol. 82, no. 3, pp. 1377–1404, 2015.

[44] P. Kumar, S.-G. Lee, and H.-J. Lee, "E-SAP: Efficient-strong authentication protocol for healthcare applications using wireless medical sensor networks," *Sensors*, vol. 12, no. 2, pp. 1625–1647, 2012.

[45] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment," *Ad Hoc Netw.*, vol. 36, pp. 152–176, 2016.
[46] W. Shi and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Int. J. Distrib. Sensor Netw.*, vol. 2013, no. 730831, pp. 51–59, 2013.
[47] W.-L. Tai, Y.-F. Chang, and W.-H. Li, "An IoT notion–based authentication and key agreement scheme ensuring user anonymity for heterogeneous ad hoc wireless sensor networks," *J. Inf. Secur. Appl.*, vol. 34, pp. 133–141, 2017.
[48] M. Turkanovic, B. Brumen, and M. Holbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion," *Ad Hoc Netw.*, vol. 20, no. 2, 2014.
[49] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, "A dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1646–1656, 2012.
[50] R. Amin, S. H. Islam, G. Biswas, M. K. Khan, and N. Kumar, "A robust and anonymous patient monitoring system using wireless medical sensor networks," *Future Gener. Comput. Syst.*, vol. 80, pp. 483–495, 2018.
[51] R. Madhusudhan and R. Mittal, "Dynamic ID-based remote user password authentication schemes using smart cards: A review," *J. Netw. Comput. Appl.*, vol. 35, no. 4, pp. 1235–1248, 2012.
[52] D. Wang, D. He, P. Wang, and C. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 4, pp. 428–442, Jul./Aug. 2015.
[53] W. Shi and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography a new user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 4, 2013, Art. no. 730831.
[54] F. Wu *et al.*, "A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks," *Future Gener. Comput. Syst.*, vol. 82, pp. 727–737, 2018.
[55] C. C. Chang and H. D. Le, "A provably secure, efficient, and flexible authentication scheme for Ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2016.
[56] P. Gope and T. Hwang, "A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7124–7132, Nov. 2016.
[57] X. Li, J. Niu, M. Z. A. Bhuiyan, F. Wu, M. Karuppiah, and S. Kumari, "A robust ECC based provable secure authentication protocol with privacy preserving for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3599–3609, Aug. 2018.
[58] X. Li, J. Peng, M. S. Obaidat, F. Wu, M. K. Khan, and C. Chen, "A secure three-factor user authentication protocol with forward secrecy for wireless medical sensor network systems," *IEEE Syst. J.*, 2019, to be published, doi: 10.1109/JSYST.2019.2899580.
[59] A. Gupta, M. Tripathi, T. J. Shaikh, and A. Sharma, "A lightweight anonymous user authentication and key establishment scheme for wearable devices," *Comput. Netw.*, vol. 149, pp. 29–42, 2019.
[60] J. Srinivas, D. Mishra, S. Mukhopadhyay, and S. Kumari, "Provably secure biometric based authentication and key agreement protocol for wireless sensor networks," *J. Ambient Intell. Humanized Comput.*, vol. 9, no. 4, pp. 875–895, 2018.
[61] L. Xiong, D. Peng, T. Peng, H. Liang, and Z. Liu, "A lightweight anonymous authentication protocol with perfect forward secrecy for wireless sensor networks," *Sensors*, vol. 17, no. 11, 2017, Art. no. 2681.
[62] F. Wu, L. Xu, S. Kumari, and X. Li, "A new and secure authentication scheme for wireless sensor networks with formal proof," *Peer Peer Netw. Appl.*, vol. 10, no. 16, pp. 16–30, 2017.
[63] J. Moon, D. Lee, Y. Lee, and D. Won, "Improving biometric-based authentication schemes with smart card revocation/reissue for wireless sensor networks," *Sensors*, vol. 17, no. 5, 2017, Art. no. 940.
[64] A. G. Reddy, A. K. Das, E.-J. Yoon, and K.-Y. Yoo, "A secure anonymous authentication protocol for mobile services on elliptic curve cryptography," *IEEE Access*, vol. 4, pp. 4394–4407, 2016.
[65] A. K. Das, S. Kumari, V. Odelu, F. Wu, F. Wu, and X. Huang, "Provably secure user authentication and key agreement scheme for wireless sensor networks," *Secur. Commun. Netw.*, vol. 9, no. 16, pp. 3670–3687, 2016.
[66] Y. Lu, L. Li, H. Peng, and Y. Yang, "An energy efficient mutual authentication and key agreement scheme preserving anonymity for wireless sensor networks," *Sensors*, vol. 16, no. 6, p. 837, 2016.
[67] Y. Choi, D. Lee, and D. Won, "Security improvement on biometric based authentication scheme for wireless sensor networks using fuzzy extraction," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 1, 2016, Art. no. 8572410.
[68] Y. H. Park, S. Y. Lee, and C. K. Kim, "Secure biometric-based authentication scheme with smart card revocation/reissue for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 7, pp. 1–11, 2016.
[69] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, "An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 76, pp. 37–48, 2016.
[70] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, L. Leng, and N. Kumar, "Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," *Comput. Netw.*, vol. 101, no. C, pp. 42–62, 2016.
[71] R. Amin and G. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Netw.*, vol. 36, no. part 1, pp. 58–80, 2016.
[72] D. He, N. Kumar, and N. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Inf. Sci.*, vol. 321, pp. 263–277, 2015.
[73] M. Turkanović, 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 Sensor Wireless Netw.*, vol. 20, pp. 96–112, 2014.
[74] M. Turkanovic and M. Holbl, "An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *Elektronika Ir Elektrotechnika*, vol. 19, no. 6, pp. 109–116, 2013.
[75] E.-J. Yoon and C. Kim, "Advanced biometric-based user authentication scheme for wireless sensor networks," *Sensors Lett.*, vol. 11, no. 9, pp. 1836–1843, 2013.
[76] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *J Netw. Comput. Appl.*, vol. 36, no. 1, pp. 316–323, 2013.
[77] P. Kumar, A. J. Choudhury, M. Sain, S.-G. Lee, and H.-J. Lee, "RUASN: A robust user authentication framework for wireless sensor networks," *Sensors*, vol. 11, no. 5, pp. 5020–5046, 2011.
[78] H.-L. Yeh, T.-H. Chen, P.-C. Liu, T.-H. Kim, and H.-W. Wei, "A secured authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 11, no. 5, pp. 4767–79, 2011.
[79] 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 Netw.*, vol. 10, no. 4, pp. 361–371, 2010.
[80] W. Li, D. Wang, and P. Wang, "Insider attacks against multi-factor authentication protocols for wireless sensor networks," *Ruan Jian Xue Bao/J. Softw.*, vol. 30, no. 8, 2019, Art. no. 2375-2391.

Chenyu Wang is currently working toward the PhD degree in the Beijing University of Posts and Telecommunications, and a visiting student in Nanyang Technological University. She has received the "Cyber security scholarship" (China) and has published several papers. Her research interests include cryptographic protocols and software security.

Ding Wang received the PhD degree in information security from Peking University, in 2017. He is currently a professor with the College of Cyber Science, Nankai University, and also serves as the deputy director of the Tianjin Key Laboratory of Network and Data Security Technology. As the first author, he has published more than 40 papers at venues like ACM CCS, Usenix Security, NDSS, IEEE DSN, ESORICS, ACM ASIACCS, ACM TCPS, *IEEE Transactions on Dependable and Secure*, and *IEEE Transactions on Information Forensics and Security*. Seven of them are recognized as "ESI highly cited papers". His PhD thesis receives the "ACM China Doctoral Dissertation Award" and "China Computer Federation (CCF) Outstanding Doctoral Dissertation Award". He has been involved in the community as a TPC, AEC member or PC Chair for more than 50 international conferences, such as Usenix Security, ACSAC, ISC, CNS, SEC, ACISP, and SocialSec. His research interests focus on authentication and provable security.

Yi Tu received the bachelor's degree from Nankai University, China, in 2016, and the master's degree from the George Washington University, in USA, in 2018. Currently he is working toward the PhD degree in the School of Physical and Mathematical Sciences in Nanyang Technological University. His research interests include machine learning, software security and symmetric key cryptanalysis.

Guoai Xu is currently a professor with the Beijing University of Posts and Telecommunications. He is a member of cyberspace security association in China. He has published more than 50 papers at venues like WWW, FSE/ESEC, and TECS. His research interests include information security, cryptographic, and software security.

Huaxiong Wang received the PhD degree in mathematics from the University of Haifa, Israel, in 1996, and the PhD degree in computer science from the University of Wollongong, Australia, in 2001. He joined Nanyang Technological University in 2006, and is currently an associate professor with the Division of Mathematical Sciences. He is also an honorary fellow at Macquarie University, Australia. His research interests include cryptography, information security, coding theory, combinatorics, and theoretical computer science. He has been on the editorial board of three international journals: *Designs, Codes and Cryptography* (2006–2011), the *Journal of Communications* (JCM), and the *Journal of Communications and Networks*. He was the program co-chair of Ninth Australasian Conference on Information Security and Privacy (ACISP 04), in 2004 and Fourth International Conference on Cryptology and Network Security (CANS 05), in 2005, and has served in the program committee for more than 70 international conferences. He has received the inaugural Award of Best Research Contribution from the Computer Science Association of Australasia, in 2004.