Efficient and Enhanced Broadcast Authentication Protocols based on Multilevel μTESLA

Xiang Li, Na Ruan*, Fan Wu, Jie Li, Mengyuan Li
Shanghai Jiao Tong University, China
* Email: naruan@cs.sjtu.edu.cn

***Abstract*—Providing lightweight authentication and resisting Denial of Service (DoS) attacks are challenging problems in wireless ad hoc networks, such as wireless sensor networks (WSNs). We introduce two improved protocols based on fault-tolerant protocol and DoS-resistant protocol in Multilevel μTESLA to overcome these difficulties. The proposed Efficient Fault-Tolerant Protocol contributes in shortening the recovery time when high-level packets are lost, and hence reduces the risk of memory-based DoS attacks. The proposed Enhanced DoS-Resistant Protocol enhances the resistance to DoS attacks by offering packet-loss recovery of authentication message.**

***Index Terms*—Broadcast Authentication Protocol (BAP); Wireless Sensor Network (WSN); Timed Efficient Stream Loss-tolerant Authentication (TESLA); Denial of Service (DoS)**

I. Introduction

A. Background

Different from traditional networks, wireless sensor networks (WSNs) have been applied to various areas such as health care, battle field and environment monitoring due to their unique characteristics, such as large-scale and real-time data dissemination, rapid deployment, self-organization, and limited requirement of resources [1–4]. However, it is these characteristics that make data transmissions meet many new challenges, such as packet loss and Denial of Service (DoS) attacks, which would make troubles in maintaining network security.

Packet loss, mainly caused by low Quality of Service (QoS) support, is becoming increasingly significant with the development of networks [5]. Traditional QoS requirements mainly result from the rising popularity of end-to-end bandwidth-hungry multimedia applications and various types of networks [6]. Different from traditional networks, several challenges and requirements have raised in WSNs due to following reasons: severe resource constraints, unbalanced traffic, data redundancy, network dynamics, scalability, etc. [7]. Surveys such as [8], [9] and [10] conclude that the packet delivery performance is fairly pessimistic in WSNs. Facing such network environment with low QoS, relevant security protocol must be designed to tolerate packet-loss in order to maintain the continuity of source authentication.

DoS attacks usually refer to that attackers repeat sending packets with different content names within a short time [11]. Severe DoS attacks have been a basic but critical challenge in network security, which may lead to severe jamming and even the paralysis of the whole sensor network [12, 14]. Memory-based attacks and computation-based attacks are two common ways of DoS attacks [13], which both are huge threats to WSNs, such as worm-hole attack, black-hole attack, and gray-hole attack [15]. Considering the limited storage and computation capability of micro processors in sensor nodes, ensuring satisfactory network security in WSNs is more difficult and demanding.

B. Previous Work

Lossy communication channels and possible DoS attacks produce many challenging problems in network security in WSNs. To meet this requirement, several broadcast authentication protocols have been proposed in recent years.

Perrig *et al.* proposed μTESLA (the micro version of TESLA), a well-known broadcast authentication protocol designed for WSNs [17]. Similar to TESLA [18], μTESLA keeps using one-way functions and symmetric cryptographic mechanism to authenticate messages, and makes some modifications to adapt to WSNs [19, 20]. However, μTESLA has some defects in some aspects as well, such as distribution of initial parameters and resisting DoS attacks.

To decrease the heavy communication overhead caused by distributing initial parameters, Liu and Ning proposed another scheme, called multilevel μTESLA [16]. In multilevel μTESLA, the single key chain used in μTESLA is replaced by multilevel key chains to shorten the average length of the key chains, while short time intervals are still used.

Besides, a solution to resist DoS attacks was proposed by Perrig *et al.* [18]. In this protocol, each packet contains hash of one or two previous packets to achieve non-repudiation. Multilevel μTESLA also uses similar idea, resisting DoS attacks by assigning commitments of high-level packets in their previous packets. Consequently, sensor nodes can authenticate packages immediately, without relying on buffers, and thus DoS attacks based on memory can be resisted.

C. Motivations

As mentioned in previous sections, multilevel μTESLA uses multilevel key chains, which yet causes many problems, such as the lack of tolerating packet loss. Although a series of schemes have been proposed to fix these problems, some critical problems still exist.

Fault-tolerant scheme is proposed to solve the problem that not all packet loss can be tolerated due to multilevel key chains. In this scheme, different levels of key chains are

978-1-4799-7575-4/14/$31.00 ©2014 IEEE

connected by another one-way function, thus low-level packet loss can be totally tolerated. However, high-level packet loss still requires long recovery time. For example, if the ith high-level package is missed during time interval I_i, all low-level packets received in time interval I_{i+2} have to be buffered for one or two high-level time intervals, which is too long for broadcasting emergent messages.

Another scheme, DoS-resistant protocol, gives one efficient way to authenticate high-level packets. Since sensor nodes do not need to buffer packets in this scheme, DoS attacks which based on memory can be resisted. However, lossy communication environment would make the scheme useless. Since the pseudorandom function used in this scheme can only connects two consecutive high-level packets, this kind of authentication lacks continuity, which means that sensor nodes cannot keep authenticating as long as one or more packets are lost.

D. Challenging Issues

As a matter of fact, these two problems, long recovery time of high- level packet loss and discontinuity of resisting DoS attacks, are difficult to solve for several reasons. Here, we briefly introduces some challenging issues, and our solutions will be discussed in following sections.

The simplest way to recover from high-level packet loss is to repeatedly send high-level packets in one high-level time interval. However, this would lead to communication overhead, and still cannot guarantee that at least one packet can be received by sensor nodes.

Meanwhile, changing the structure of multilevel key chains may possibly affect the security of this mechanism. In other words, when related high-level or low-level keys are used to shorten the recovery time, the protocol must ensure that these keys cannot be utilized by attackers.

Analogously, to make packet loss tolerated in DoS-resistant scheme, such recovery mechanism we need to establish should be as secure as the original one. This requires that the authentication process is efficient and elegant to guarantee that the recovery mechanism leaves no chance for attackers to forge packets.

E. Our Work

To solve two problems mentioned before, we present two protocols improved from multilevel μTESLA, namely Efficient Fault-Tolerant Protocol (EFTP) and Enhanced DoS-Resistant Protocol (EDRP). We present these two protocols because they both enhance resistance to DoS attacks by efficiently tolerating packet loss.

Efficient Fault-Tolerant Protocol shortens the recovery time of lost packets. Specifically, when high-level packets are lost, using our protocol can shorten the recovery time by one high-level time interval. Since time needed for buffering packets before authentication can be shortened, DoS attacks based on memory can also be mitigated. We provide evaluation of this protocol.

Enhanced DoS-Resistant Protocol contributes in tolerating packet loss. In other words, when one or more packets are lost, while DoS-resistant mechanism will no longer take effects in original scheme, it can still take effects in our protocol, which is especially meaningful when communication channels are lossy. Simulation results will be presented.

In the rest of this paper, Section II briefly introduces the relevant background knowledge, including several broadcast authentication protocols on which our work based . In section III and section IV, we completely describe two protocols we present, the Efficient Fault-Tolerant Protocol and the Enhanced DoS-Resistant Protocol, respectively. In section V, we show the evaluation of our work. We will draw a conclusion in Section section VI.

II. Preliminaries

In severely resource-constrained environments like WSNs, TESLA (Timed Efficient Stream Loss-tolerant Authentication) [18], μTESLA [17] and multilevel μTESLA [16] have been successively designed to achieve broadcast authentication. Our work is based on these three protocols to solve corresponding problems occurred in these protocols.

A. Notation

Here, we give some notations of variables:

- I_i: the ith time interval
- k_i, K_i: the shared secret key used in time interval I_i
- $k_{i,j}$, $K_{i,j}$: the jth key used in time interval I_i
- F: one-way hash function used to generate keys
- d: number of time intervals of key disclosure delay
- $P_{i,m}$: the mth packet received in time interval I_i
- $MAC_{K_i}(M)$: MAC computed by encrypting message M with key K_i

B. TESLA, μTESLA

Generally speaking, μTESLA, which is modified from TESLA, is widely applied to authentication of broadcasting messages in WSNs. In traditional networks, asymmetric cryptographic mechanisms, such as digital signature scheme, are implemented in case attackers may send forged packets to receivers. However, the high communication and computation overhead caused by digital signatures are not practical in severely resource-constrained sensor network [4]. Instead, TESLA and μTESLA are proposed to use symmetric cryptography to achieve asymmetric property, taking advantage of sender's delayed disclosure of keys.

The main idea of TESLA is that each packet is attached with a message authentication code (MAC), which is computed with a shared secret key, k_i, over the contents of the packet, i.e. $MAC_{K_i}(Message)$. In addition, keys are derived from a one-way key chain, $k_i = F(k_{i+1})$, where F is a one-way function, which implies k_{i+1} cannot be derived from k_i even F is known. Specifically, each key k_i is used in the time interval I_i, and would be disclosed after (d – 1) time intervals to make keys secret during this period of time.

For the receiver, each packet with attached MAC should be buffered as long as corresponding key is still secret. When the key is disclosed after (d – 1) time intervals, the receiver can use disclosed key to compute the theoretical MAC of the buffered packet and then compare it with the MAC attached. If two MACs are same, this packet would be accepted. Otherwise, this packet may be forged and needs to be abandoned.

Besides to the asymmetric property, TESLA also provides the property of tolerating packet loss. Since each key is derived from a one-way key chain, the receiver can use k_{i+1}, which is disclosed in the time interval I_{i+1+d}, and the one-way function F to compute k_i if k_i is lost, by relation $k_i = F(k_{i+1})$.

However, TESLA is not initially designed for environments such as sensor networks whose resources (storage and computing capability) are so limited. On the contrary, μTESLA, which is also proposed by Perrig *et al.*, improved TESLA in several aspects as follows: μTESLA uses symmetric mechanisms instead of digital signature in authentication of the initial packet; instead of disclosing a key in each packet in TESLA, μTESLA discloses the key once per epoch, avoiding communication overheads caused by excess key disclosures [13].

Figure 1 gives an example of μTESLA. One-way function F_0 is used to generate K_i. In each time interval I_i, packets from $P_{i,1}$ to $P_{i,n}$ share the same key K_i to compute respective MAC.

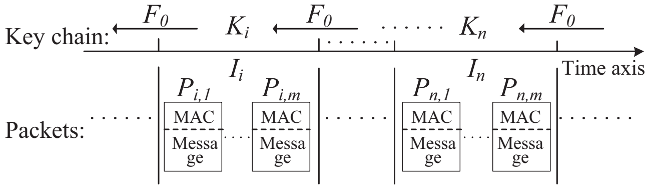


Fig. 1: Keys' generation and usage in μTESLA

C. Multilevel μTESLA

Although μTESLA improves the method of initialization in TESLA to accommodate resource-constraint wireless sensor networks, initializing sensor nodes by unicasting initial parameters in μTESLA costs a lot of time, which may not scale up to large sensor networks with thousands of networks [16]. Multilevel μTESLA is designed to solve this problem and presents a series of schemes to keep the nice properties of μTESLA. To predetermine and broadcast the initial parameters, a multilevel key chain scheme is used, where the high-level key chains are used to authenticate the commitments ($K_{i,0}$) of lower-level key chains. In fact, high-level key chains are used in high-level packets, namely *commitment distribution message (CDM$_i$)*, which is composed as follows:

$$CDM_i = i \mid K_{i+2,0} \mid MAC_{K_i'}(i \mid K_{i+2,0}) \mid K_{i-1}$$

CDM_i consists of the commitment $K_{i+2,0}$ together with corresponding MAC for the low-level key chain, and the high-level key K_{i-1} to authenticate the high-level packets [16]. Take two-level μTESLA as an example, two pseudorandom one-way functions, F_0 and F_1, are used to generate key chains, K_i for high-level packets, and $K_{i,n}$ for low-level packets, respectively. As a matter of fact, high-level key chain is actually designed to distribute key chain commitments of the low-level key chains to use multiple low-level key chains. Hence, multilevel μTESLA initially contributes in combining multiple key chains together.

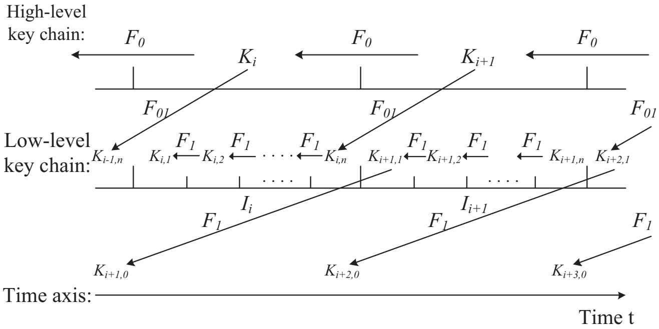


Fig. 2: Generation and usage of keys in multilevel μTESLA

Figure 2 depicts the deployment of key chains of multilevel μTESLA. Two one-way functions, F_0, F_1 are used in high-level and low-level key chains respectively. F_{01} is the one-way function to set up the connection between the high-level key chain and low-level key chains. Correspondingly, K_i and $K_{i,j}$ are keys generated in high-level key chain and low-level key chain, respectively.

This scheme has following advantages: the high-level key chain can cover a long period of time without having a too long key chain due to long time interval; the low-level key chain provides short key chain intervals to avoid the high demand of computation and storage resource of long key chains.

Next, we will focus on two schemes proposed in multilevel μTESLA. Fault-tolerant scheme is designed to maintain the property of tolerating packet loss in μTESLA. In specific, K_{i,n_1} is furthermore connected to K_{i+1} by another pseudorandom one-way function, F_{01}, with $K_{i,n} = F_{01}(K_{i+1})$, where $K_{i,n}$ is the last low-level key in the time interval I_i. Consequently, loss of the last key $K_{i,n}$ in one low-level key can be solved by high-level keys. While this improvement solves the possible loss of the last packet in the low-level interval, high-level packet loss still remains a problem, and simply repeatedly broadcast CDM_i in each time interval is not an optimal choice, which will be discussed later.

Another scheme, DoS-resistant scheme, is designed to completely defeat DoS attacks. In specific, the image of CDM_{i+1}, $H(CDM_{i+1})$, is added into CDM_i, where H is a pseudorandom function. Thus, when CDM_{i+1} is received, it can be immediately authenticated without buffering it by authenticating $H(CDM_{i+1})$ which has been received and authenticated along with CDM_i. Therefore, sensor nodes can immediately resist the memory-based DoS attacks. On the other hand, the shortcoming of this scheme is that it will become useless as soon as one CDM_i is lost, due to the lack of continuity in the

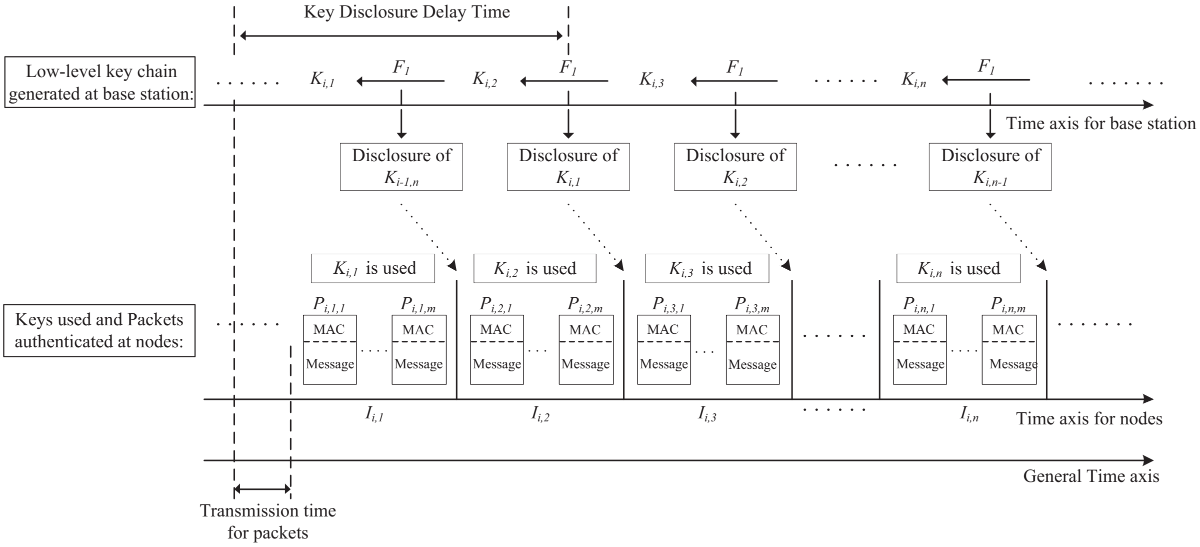


Fig. 3: Specific implementation of low level key chains

authentication process.

III. Efficient Fault-Tolerant Protocol

A. Protocol Description

We propose an improved protocol, namely Efficient Fault-Tolerant Protocol, which reconstructs the connection between high-level and low-level key chains to solve the problem in origin scheme in multilevel μTESLA. In multilevel μTESLA, though it connects low-level key chain and high-level key chain to tolerate low-level packet loss, high-level packet loss remains a problem. Our protocol contributes in shortening the recovery time from high-level packet loss.

Figure 3 shows a comprehensive process of the specific implementation of low-level key chains in Efficient Fault-Tolerant Protocol. Here, we assume that the *disclosure delay time* is 2 intervals, i.e., $d = 2$. In other words, while $K_{i,1}$ is distributed in $I_{i,1}$, it is not disclosed until $I_{i,3}$. Consequently, the sensor nodes need to buffer packets for two time intervals and then authenticate them with keys disclosed later and the MAC attached in packets.

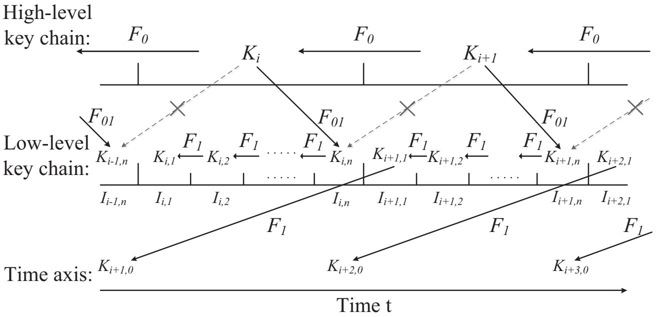


Fig. 4: Generation and usage of keys in EFTP

Figure 4 depicts the new two-level key chains, with usage and generation of key chains in Efficient Fault-Tolerant Protocol. Different from original scheme depicted in Figure 2, the one-way function F_{01} is used to connect K_i and $K_{i,n}$, instead of K_{i+1} and $K_{i,n}$, by relation $K_{i,n}$ = $F_{01}(K_i)$. The dash line represents the original connection, while the solid line represents the new connection.

In multilevel μTESLA, the main function of CDM_i is to distribute $K_{i+2,0}$, which can be used as the key chain commitment to authenticate low-level packets in time interval I_{i+2}. How $K_{i+2,0}$ is derived through CDM_i, including the case when no CDM_i is received in time interval I_i, is described in the pseudocode of algorithm 1.

To understand this algorithm, recall that CDM_i contains i, $K_{i+2,0}$, $MAC_{K'_i}(i|K_{i+2,0})$, and K_{i-1}. Line 3 to 10 take effects in the normal case when CDM_{i+1} is not lost, and sensor nodes can use K_{i-1} to authenticate K_i. Since K'_i is derived from K_i, sensor nodes can furthermore use K'_i to authenticate $K_{i+2,0}$ and its MAC. On the other hand, line 11 to 22 take effects in the case when CDM_{i+1} is lost. As long as the sensor node receives CDM_{i+3}, sensor nodes can use F_{01} to derive $K_{i+2,n}$. Then, F_1 can be used to derive $K_{i+2,0}$.

B. Improvement Analysis

As mentioned before, high-level packet loss leads to some problems in original scheme in multilevel μTESLA. When all copies of CDM_i are missed during I_i, all low-level packets received in I_{i+2} have to be buffered. Since the sensor node can use F_{01} to compute $K_{i+2,n}$ after receiving K_{i+3} during I_{i+4} (recall that CDM_i contains K_{i-1}). Then, F_1 could be used to compute $K_{i+2,0}$. Consequently, packets received in I_{i+2} have to be buffered until time interval I_{i+4}. Therefore, this buffered period of time is actually at least one high-level time interval and perhaps two high-level time intervals in the worst case, which is not a short time, especially when the key management mechanism has more than two levels.

By contrast, in Efficient Fault-Tolerant Protocol, when

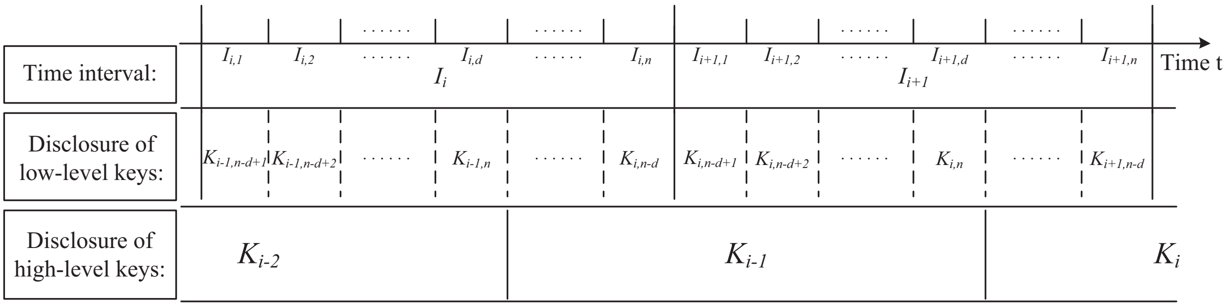


Fig. 5: Disclosure and authentication of keys in EFTP

Algorithm 1 Efficient Fault-Tolerant Protocol

Require: High-level packets *CDM*; one-way functions F_0, F_1, F_{01}; high-level key K_i; low-level key $K_{i,k}$; time interval I_i.

Ensure: Derive and authenticate $K_{i+2,0}$.

```
$CDM_i$ is received in $I_i$
buffer $CDM_i$;
if $CDM_{i+1}$ is received in $I_{i+1}$ then
    if $K_{i-1}$ == $F_0$(K$_i$) then
        derive $K'_i$ from $K_i$;
        if MAC == $K'_i(i \mid K_{i+2,0})$ then
            $K_{i+2,0}$ is authenticated;
        end if
    end if
end if
if no $CDM_{i+1}$ is received in $I_{i+1}$ then
    wait until $CDM_{i+3}$ is received;
    if $K_{i-1}$ == $F_0(F_0(F_0(K_{i+2})))$ then
        derive $K_{i+2,n}$ = $F_{01}(K_{i+2})$;
        k = n - 1;
        while k > 1 do
            $K_{i+2,k}$ = $F_1(K_{i+2,k+1})$;
            $k$ = $k$ - 1;
        end while
        K$_{i+2,0}$ is derived;
    end if
end if
```

CDM_i is lost, sensor nodes are still able to use function F_{01} to derive $K_{i+2,n}$ from K_{i+2}, which is received in CDM_{i+3} in time interval I_{i+3}. As a consequence, the recovery time for low packets received in time interval I_{i+2} is at most one high-level time interval.

C. Security Analysis

The security of this protocol is inherited from TESLA and multilevel μTESLA. The high-level key chain is predetermined in the initialization of sensor nodes with one-way function F_0, and its main function is to assist broadcasting commitments for low-level key chains (such as $K_{i+2,0}$). When the sensor node receives a CDM_i, it waits for CDM_{i+1} to get K_i. With K_{i-1} in CDM_i and F_0, K_i can be authenticated and then used to derive K'_i. After deriving K'_i, it can be used to furthermore authenticate $K_{i+2,0}$ with corresponding MAC contained in CDM_i. If authentication succeeds, $K_{i+2,0}$ can be used as the commitment of the low-level key chain during interval I_{i+2}. If any one of these two authentication fails, this implies that attackers have forged message.

As soon as $K_{i+2,0}$ is authenticated, the sensor node can use it with one-way function F_1 together to authenticate following messages with their MACs in this time interval. Given that K_i and $K_{i.n}$ are directly connected in Efficient Fault-Tolerant Protocol, in case attackers may K_i and F_{01} to compute $K_{i,n}$ and then forge messages, the disclosure time of K_i should be after the time interval $I_{i+1,d}$, which is the disclosure time of $K_{i,n}$.

Figure 5 illustrates the disclosure time of keys, the disclosure of high-level packets is changed due to the reconstruction of F_{01}. This is used in case attackers may use disclosed K_i to derive $K_{i,n}$ before $K_{i,n}$ is disclosed. In addition, considering the the key disclosure delay, K_i should be disclosed after time interval $I_{i+1,d}$. Since K_i is distributed in CDM_{i+1}, postponing the disclosure time of K_i actually postpones the distribution of CDM_{i+1}. As a result, the Efficient Fault-Tolerant Protocol requires that CDM_{i+1} could not be sent to sensor nodes until time interval $I_{i+1,d}$, which is still realistic.

IV. Enhanced DoS-Resistant Protocol

A. Protocol Description

To fix the problem of packet loss in original DoS-resistant scheme in multilevel μTESLA, we introduce another new protocol, Enhanced DoS-Resistant Protocol. In this protocol, when sensor nodes fail to receive CDM_i, the sensor nodes can shorten the recovery time of CDM_i by taking advantage of high-level key chain, K_i.

Figure 6 describes the new high-level key chain in the Enhanced DoS-Resistant Protocol. Here, *H(CDM$_i$)* and K_i are pointed out to emphasize their utility and connection. As mentioned before, *H(CDM$_i$)* is the image of CDM_i, with pseudorandom function *H*. Besides, K_i is the high-level key chain, generated by one-way function F_0.

The pseudocode of algorithm 2 is shown as follows. In line 3 to 4, *H(CDM$_i$)* is used to generate the image of CDM_{i+1},

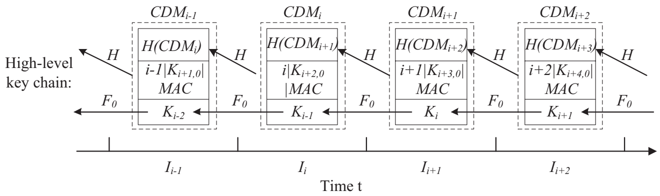


Fig. 6: Construction of high-level packets in EDRP

and then compare it with the *H(CDM*$_{i+1}$*)* contained in CDM_i. Different from the original scheme in multi-level μTESLA, line 8 to 13 take effects when high-level packet, CDM_{i+1}, is lost. In this case, high-level keys including K_{i-1} and one-way function F_0 can be used to tolerate packet loss.

Algorithm 2 Enhanced DoS-Resistant Protocol

```
Require: High-level packets $CDM$ is authenticated; one-way
    functions $F_0$; pseudorandom function $H$; high-level key
    $K_i$; time interval $I_i$.
Ensure: Authenticate $CDM_{i+1}$ or next following high-level
    packets.
 1: $CDM_i$ is received in $I_i$ and authenticated;
 2: if $CDM_{i+1}$ is received in $I_{i+1}$ then
 3:     calculate $H(CDM_{i+1})$ by $H$;
 4:     if $H(CDM_{i+1})$ in $CDM_i$ == $H(CDM_{i+1})$ then
 5:         $CDM_{i+1}$ is authenticated;
 6:     end if
 7: end if
 8: if no $CDM_{i+1}$ is received in $I_{i+1}$ then
 9:     wait until $CDM_{i+2}$ is received;
10:     if $K_{i-1} == F_0(F_0(K_{i+1}))$ then
11:         $CDM_{i+2}$ is authenticated;
12:     end if
13: end if
```

B. Improvement Analysis

In original DoS-resistant scheme, recovery from loss of CDM_i would lead to the loss of efficacy of resistance to DoS attacks. This is due to the fact that the number of images of following high-level packets contained in CDM_i is finite. Thus, loss of one or several continuous high-level packets would make the whole protocol loss efficacy.

On contrary, since high-level key chain K_{i-1} is used when CDM_i is lost in Enhanced DoS-Resistant Protocol, the resistance to DoS attacks maintains continuity when packet loss happened. In other words, take line 8 to 13 in the pseudocode as an example, even CDM_{i+1} is lost, CDM_{i+2} can still be authenticated by one-way function F_0.

C. Security Analysis

Now we begin to discuss why applying Enhanced DoS-Resistant Protocol wouldn't change the security of multilevel μTESLA. Since when high-level packet loss does not happen, the authentication mechanism remains same, our focus is on how security property is retained when recovery mechanism is used in our new protocol.

If CDM_i is not received in time interval I_i, which means the image of CDM_{i+1}, *H(CDM*$_{i+1}$*)*, is not received, the sensor node can still authenticate CDM_{i+1} by waiting until CDM_{i+2} is received. Since K_{i+1} is contained in CDM_{i+2}, the sensor node can calculate F_i by one-way function F_0 and then compare it with the value contained in CDM_{i+1}. Meanwhile, since the image of CDM_{i+2}, *H(CDM*$_{i+2}$*)*, should be contained in CDM_{i+1}, the sensor node can also re-authenticate CDM_{i+2}. If these two authentication succeed, CDM_{i+1} and CDM_{i+2} can be regarded as reliable.

On the other hand, due to the alternative authentication method offered by the new protocol, attackers may find another method to attack the sensor network. First, they may temporarily jam the communication channel for sensor networks to make high-level packets such as CDM_i lost to activate the authentication of one-way key chain. Then, if they somehow manage to capture one sensor node or to intercept one CDM_{i+1}, they would be capable of replacing the message contained in CDM_{i+1} while retaining K_{i+1} and *H(CDM*$_{i+2}$*)* and thus successfully forge CDM_{i+1}.

Although the process mentioned above is not easy to implement, we do prefer to take it into consideration. Nevertheless, the truth is that the similar case also happens without our modification of multilevel μTESLA. In original scheme, when attackers intercepts CDM_i, they can replace *H(CDM*$_{i+1}$*)* with the image of their forged CDM_{i+1}. In this case, attackers can deceive the sensor nodes by forging CDM_{i+1} instead of CDM_i. In fact, this attack can be accomplished relatively easier when comparing with the similar attack to Enhanced DoS-Resistant Protocol.

In summary, while Enhanced DoS-Resistant Protocol offers the continuity to resist DoS attacks, it still maintains same performance in security realm as the original scheme.

V. Evaluation

We have implemented the Efficient Fault-Tolerant Protocol (EFTP) in section III and the Enhanced DoS-Resistant Protocol (EDRP) in section IV. Improvement analysis and security analysis show their efficiency and reinforcement. In this section, we described our evaluation results. We ran all the test program on a PC with an Inter(R) Core(TM) i5-3317U 1.70GHz CPU and an 8.00GB RAM.

A. Evaluation Settings

In consistent with schemes in multilevel μTESLA, the duration of each low-level time interval is 100 ms, and each low-level key chain consists of 600 keys. The duration of each time interval for the high-level key chain is 60 s. According to [16], the number of high-level keys does not have an obvious impact on the performance measures. We put 200-2000 keys in the high-level key chain as confidence interval, which covers up to 200-2000 minutes in time. We also set the data packet rate at base station to 100 data packets per minute.

Every time as distribute the high-level key, attacker keeps consistent with the based station on sending packets and buffers for the CDM_i. The percentage of forged CDM packets is almost twice as the true packets and three times as the CDM_i buffers. The ration of received truth packet is 50%-100% as confidence interval. In order to ensure successful transmission, CDM_i may be sent several times. We assume its times is 0-100.

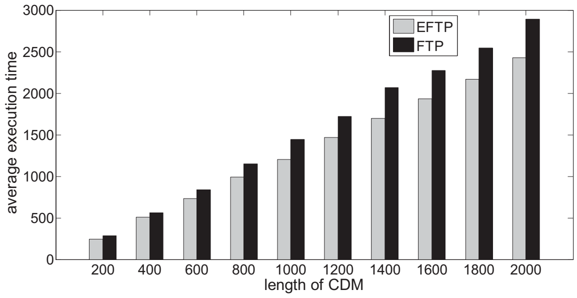


Fig. 7: Compare of average time delay

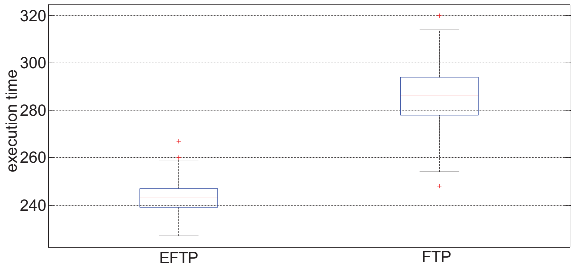


Fig. 8: Execution Time

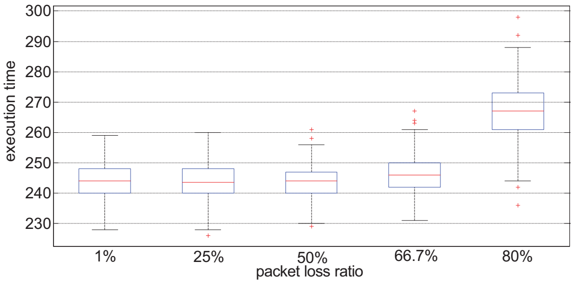


Fig. 9: Impact of packet loss rate on EFTP

B. Simulation Results

Figure 7 compares the EFTP scheme with original fault-tolerant protocol (FTP) in multilevel μTESLA. Horizontal ordinate indicates the length of CDM, while vertical ordinate is the execution time of each CDM. According to the figure, our scheme reduces the delay time as providing the same authentication. As increase of the length of key-chain in the high-level, both of the schemes cost more time for correct authentication. However, the cost of our scheme increases slowly and is much less than the FTP in multilevel μTESLA.

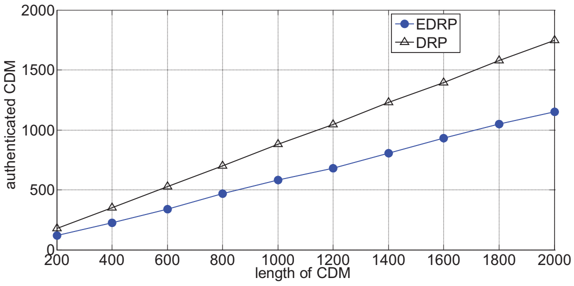


Fig. 10: Impact on number of CDM for authentication

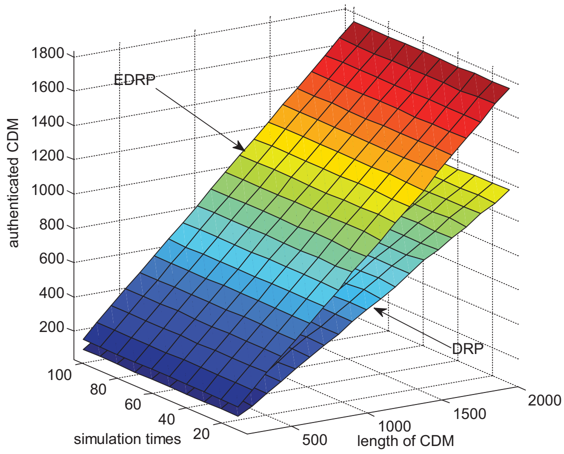


Fig. 11: Average of impact on number of CDM

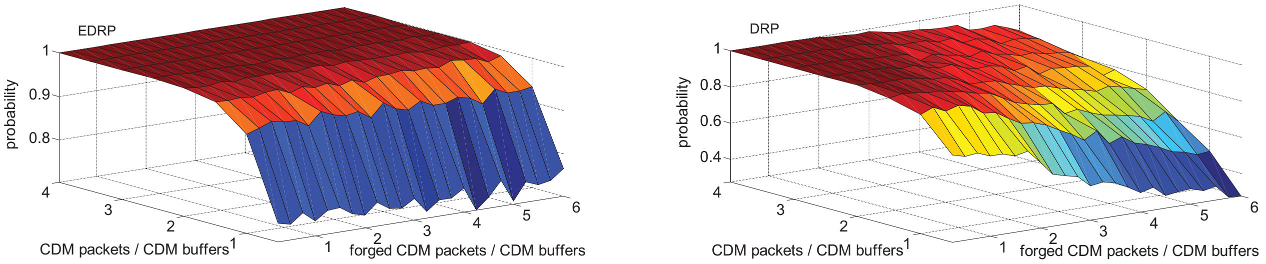
Figure 8 and Figure 9 are Box-plots. Figure 8 shows EFTP is more stable than FTP and cost less execution time. Figure 9 demonstrates the impact of packet loss rate to EFTP. If the packet loss rate is less than 50%, EFTP can provide a low and stable execution time. As we know, packet loss rate in wireless communication is often up to 90%, which means EFTP can serve the wireless network very well.

Figure 10 and Figure 11 compare the EDRP scheme with original DoS-resistant protocol (DRP) in multilevel μTESLA. In both of the simulations, we study how length of key chain in high-level effects the schemes. In Figure 10, we see the correct authenticated key in high-level increased if the key-chain is longer enough. At the same time, our scheme has higher ration of the correct key numbers. Simulation times in Figure 11 shows our scheme still has advantage under the average situations.

Figure 12 shows the impact of the storage space on the number of high-level keys. Compared with DRP, our scheme reduces the impact as increasing forge packets.

C. Summary and Future Work

In summary, our evaluation results show that our protocols, EFTP and EDRP, have better performance than original schemes. On the other hand, our evaluation can still be improved. For example, different from our assumption, there may exist multiple base stations, one of which may be



(a) EDRP

(b) DRP in Multilevel-μTESLA

Fig. 12: Impact of the storage space utilization

compromised by attackers in some scenarios. We will give a more complicated evaluation in further research.

VI. Conclusion

We have designed and evaluated two lightweight broadcast authentication protocols. Our protocols are superior to previous work in their ability to provide efficient fault-tolerance and enhanced DoS-resistance. We show that these two broadcast authentication protocols can mitigate the overhead and improve the correct authentication ration. The protocols we have proposed in this paper enable resource-constrained devices to verify messages efficiency, which is important for the connection between different networks as development of Internet of Things (IoTs).

Acknowledgement

This work was supported in part by CCF-Tencent Open Fund and CCF-Intel Young Faculty Researcher Program, in part by China NSF grant 61422208, 61272443 and Shanghai Jiao Tong University 211 Fund.

References

[1] X. Fan, and G. Gong, *Accelerating signature-based broadcast authentication for wireless sensor networks*, Ad Hoc Networks, vol. 10, no.4, pp. 723-736, June 2012.

[2] K. Shim, Y. Lee , and C. Park, *EIBAS: An efficient identity-based broadcast authentication scheme in wireless sensor networks*, Ad Hoc Networks, vol. 11, no. 1, pp. 182-189, Jan. 2013.

[3] K. Ren, S. Yu, W. Lou, and Y. Zhang, *Multi-User Broadcast Authentication in Wireless Sensor Networks*, IEEE Transactions on Vehicular Technology, vol. 58, no. 8, pp. 4554-4564, Oct. 2009.

[4] S. Yamakawa, Y. Cui, K. Kobara, and H. Imai, *Lightweight broadcast authentication protocols reconsidered*, in Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Budapest, Hungary, pp. 1-6, April 2009.

[5] M. Fiedler, T. Hossfeld, and P. Tran-Gia, *A generic quantitative relationship between quality of experience and quality of service*, Network, IEEE, vol. 24, no. 2, pp. 36-41, March 2010.

[6] D. Chen, and P. K. Varshney, *QoS Support in Wireless Sensor Networks: A Survey*, in Proceedings of the 2004 International Conference on Wireless Networks (ICWN), Las Vegas, Nevada, USA, vol. 233, pp. 1-7, June 2004.

[7] K. C. Rahman, *A survey on sensor network*, Journal of Computer and Information Technology, vol. 1, no. 1, pp. 76-87, 2010.

[8] M. A. Yigitel, O. D. Incel, and C. Ersoy, *QoS-aware MAC protocols for wireless sensor networks: A survey*, Computer Networks, vol. 55, no. 8, pp. 1982-2004, June 2011.

[9] J. Li, C. Blake, D. S. J. D. Couto, H. Lee, and R. Morris, *Capacity of ad hoc wireless networks*, in Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom), Rome, Italy, pp. 61-69, July 2001.

[10] M. Natkaniec, K. Kosek-Szott, S. Szott, and G. Bianchi, *A survey of medium access mechanisms for providing qos in ad-hoc networks*, Communications Surveys & Tutorials, IEEE, vol. 15, no. 2, pp. 592-620, May 2013.

[11] S. Choi, K. Kim, S. Kim, and B. H. Roh, *Threat of DoS by interest flooding attack in content-centric networking*, in Proceedings of the IEEE International Conference on Information Networking (ICOIN), Bangkok, Thailand, pp. 315-319, Jan. 2013.

[12] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy. *Denial of service attacks in wireless networks: The case of jammers*, Communications Surveys & Tutorials, IEEE, vol. 13, no. 2, pp. 245-257, May 2011.

[13] A. Studer, F. Bai, B. Bellur, and A. Perrig, *Flexible, Extensible, and Efficient VANET Authentication*, Journal of Communications and Networks, vol. 11, no. 6, pp. 574-588, Dec. 2009.

[14] D. R. Raymond, and S. F. Midkiff, *Denial-of-service in wireless sensor networks: Attacks and defenses*, Pervasive Computing, IEEE, vol. 7, no. 1, pp. 74-81, Jan. 2008.

[15] R. H. Jhaveri, S. J. Patel, and D. C. Jinwala, *DoS attacks in mobile ad hoc networks: A survey*, in Proceedings of Second International Conference on Advanced Computing & Communication Technologies (ACCT), Panipat, India, pp. 535-541, Jan. 2012.

[16] D. Liu, and P. Ning. *Multilevel μTESLA: Broadcast authentication for distributed sensor networks*, ACM Transactions on Embedded Computing Systems (TECS), vol. 3, no. 4, pp. 800-836, Nov. 2004.

[17] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Cullar, *SPINS: Security protocols for sensor networks*, Wireless Networks, vol. 8, no. 5, pp. 521-534, Sept. 2002.

[18] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, *Efficient authentication and signing of multicast streams over lossy channels*, IEEE Symposium on Security and Privacy, Berkeley, CA, USA, pp. 56-73, May 2000.

[19] H. Tan, J. Zic, S. K. Jha, and D. Ostry, *Secure multihop network programming with multiple one-way key chains*, in Proceedings of IEEE Transactions on Mobile Computing, vol. 10, no. 1, pp. 16-31, Jan. 2011.

[20] Q. Yu, and C. N. Zhang, *Using RC4-BHF to Construct One-way Hash Chains*, in Proceedings of the 3rd International Conference on Computer Engineering and Network (CENet), Shanghai, China, vol. 277, pp. 1133-1141, Dec. 2013.

[21] H. C. Hsiao, A. Studer, C. Chen, A. Perrig, F. Bai, B. Bellur, and A. Iyer, *Flooding-resilient broadcast authentication for VANETs*, in Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom), Las Vegas, Nevada, USA, pp. 193-204, Sept. 2011.