2007

Containing denial-of-service attacks in broadcast authentication in sensor networks

Ronghua Wang
Syracuse University, rwang01@ecs.syr.edu

Wenliang Du
Syracuse University, wedu@ecs.syr.edu

Peng Ning
North Carolina State University at Raleigh, pning@ncsu.edu

Containing Denial-of-Service Attacks in Broadcast Authentication in Sensor Networks

Ronghua Wang, Wenliang Du *
Department of EECS
Syracuse University
{rwang01, wedu}@ecs.syr.edu

Peng Ning †
Department of Computer Science
North Carolina State University
pning@ncsu.edu

ABSTRACT

Broadcast authentication is an important application in sensor networks. Public Key Cryptography (PKC) is desirable for this application, but due to the resource constraints on sensor nodes, these operations are expensive, which means sensor networks using PKC are susceptible to Denial of Service (DoS) attacks: attackers keep broadcasting bogus messages, which will incur extra costs, thus exhaust the energy of the honest nodes. In addition, the long time to verify each message using PKC increases the response time of the nodes; it is impractical for the nodes to validate each incoming message before forwarding it.

In this paper we discuss this type of DoS attacks, in which the goal of the adversary is to exhaust the energy of the sensor nodes and to increase their response time to broadcast messages. We then present a dynamic window scheme, where sensor nodes determine whether first to verify a message or first to forward the message by themselves. This is made possible with the information such as how far this node is away from the malicious attacker, and how many hops the incoming message has passed. We compare the performance of the proposed scheme with other schemes, and show that it can contain the damage of DoS attacks to only a small portion of the sensor nodes.

1. INTRODUCTION

Sensor networks are being deployed for a wide variety of applications, such as military sensing and tracking, etc. A typical sensor network usually has one or more base stations that serve as the commanders and data sinks. They broadcast commands to sensors, which act upon those commands. Due to the large geographical dimensions that sensor networks are deployed, broadcast is often achieved in the relay fashion: intermediate nodes forward messages to nodes that cannot hear the base stations directly.

Adversaries may impersonate base stations if the authenticity of the broadcast messages are not guaranteed. Broadcast authentication can be used to ensure the authenticity, and several broadcast authentication schemes in sensor networks have been proposed. For example, μTesla [9] is an efficient broadcast authentication scheme based on one-way hash chain. However, it can only achieve a delayed authentication, which is undesirable for time-sensitive broadcast messages. In contrast, public-key-based signature schemes can achieve real-time broadcast authentication, but their operations are expensive in terms of energy consumption and running time. This makes the public-key-based broadcast authentication schemes in sensor networks susceptible to Denial of Service (DoS) attacks. To launch DoS attacks, malicious nodes can keep broadcasting meaningless messages; upon receiving these faked messages, if sensor nodes forward them to their neighbors before they authenticate the messages (we call it the *forwarding-first* method), the faked messages will be spread across the entire sensor networks, consuming sensors' energy. Although sensors will eventually drop the faked messages after the verification fails, the damage has already been made.

A straightforward way to deal with this type of attacks is to verify each message before forwarding it (we call it the *authentication-first* method). The faked messages will be dropped at the first-hop neighbors of the malicious nodes, so nodes beyond them will not be affected. Although this is preferable when dealing with faked messages, it has significant penalty on legitimate broadcast messages, because it takes time for sensor nodes to conduct message authentication. For example, signature verification using 160-bit elliptic curve keys on ATmega128, a processor used in Mica motes, may take as much as 1.6 seconds [5]. If every node verifies the incoming packets before forwarding them, there will be a long delay for remote nodes to obtain an authentic message. For time-sensitive broadcast messages, this is not affordable.

A desirable property of conducting authentication before forwarding is, no faked broadcast messages will be propagated, which is desirable for tolerating DoS attacks. An ideal solution is to conduct authentication-first for faked messages, and forwarding-first for authentic ones. However, this is hard to achieve, because sensor nodes have no idea on whether they are first hop victims of the attackers or not. In this paper, we propose a dynamic window scheme that is the combination of the authentication-first and the forwarding-first scheme, which can achieve a good trade-off between the broadcast delay for authentic messages and energy savings for faked messages.

The basic idea of our scheme is that, sensor nodes gradually shift to authentication-first scheme if they start receiving many faked messages, but will remain in forwarding-first mode if the majority of the messages they receive are authentic. The decision is based on the validity of the incoming broadcast messages they receive.

*Du's work is supported by Grant CNS-0430252 from the US National Science Foundation and also by Grant W911NF-05-1-0247 from the US Army Research Office.

†Ning's work is supported by Grant CAREER-0447761 from the US National Science Foundation.

MobiHoc'07, September 9–14, 2007, Montréal, Québec, Canada.

Every broadcast message keeps record of the number of hops it has passed since its last authentication, and sensor nodes maintain an authentication window size, which will be updated dynamically. Based on both the window size on sensor nodes and the number of hops the incoming message passes after its last authentication, the nodes decide which mode to use: if window size is the larger, they use forwarding-first mode; otherwise, they use authentication-first mode. In our scheme, we use Additive Increase Multiplicative Decrease (AIMD) techniques to dynamically manage the window size on sensor nodes: if the message they receive is authentic, the window size increases; otherwise, window size decreases.

Specifically, we make the following contributions in our paper:

Design: Our dynamic window scheme is an efficient yet effective protocol that can contain the damage of DoS attacks to a small portion of the sensor nodes. AIMD itself is not a new idea; it has been used in congestion control in sensor networks as well as in general networks. However, designing a DoS resistant scheme for broadcast authentication in sensor networks is not a trivial extension of previous works: sensor nodes have no idea on who is malicious and who is not. What is more, sensor nodes are extremely resource-constrained, and they should not be carried away by the overwhelming attacks from the adversaries. The design of this DoS resistant scheme is an important contribution of this paper.

Analysis: We analyze the various patterns of DoS attacks the adversaries may implement, and evaluate the performance of the proposed scheme under these attacks. The analysis may also be extended to other applications, which is an important contribution of this paper. We also validate these analysis with experiments.

Organization The organization of our paper is as the following: Section 2 discusses the related works, followed by the description of the system model and design goal of the scheme. In Section 4, we present our scheme (a dynamic window scheme) and its properties. This is followed by the evaluation and analysis of our scheme in Section 5. Finally, Section 6 concludes the paper.

2. RELATED WORKS

DoS attacks are very serious threats to the resource-constrained sensor networks. Wood and Stankovic summarized the various DoS attacks against sensor networks in [14]. McCune et al. [6] proposed a secure implicit sampling scheme to detect DoS attacks in sensor networks, where base stations probabilistically request authenticated acknowledgment from a subset of nodes per broadcast. However, for the attacks discussed earlier, broadcast messages still reach the intended receivers, so the attack is still difficult to detect. Deng et al. [2] proposed using a one-way hash chain to protect end-to-end communication in sensor networks against path-based DoS attacks, but the proposed solution cannot handle the DoS attacks described previously either.

Broadcast authentication is used in sensor network to prevent the attackers from impersonating the base stations. Previous broadcast authentication schemes in sensor networks focus primarily on symmetric keys. For example, μTESLA, proposed by Perrig et al. [9], is based on one-way hash chain of commitments. It is resilient to packet loss and has low communication overheads, but receivers cannot verify signature instantly. In [8], μTESLA was extended to an immediate authentication mechanism by replacing receiver buffering with sender buffering, but it is not desirable for applications where broadcast commands cannot be predicated in advance. These shortcomings make public key operations desirable for broadcast authentication, but the high costs of public keys used to limit the usage of public keys in sensor nodes [9]. Recently, studies show that public keys are feasible in sensor networks, especially with the Elliptic Curve Cryptography (ECC). For example, [5] points out that signature verification can be done in 1.6 seconds with 160-bit ECC keys on ATmega128 8-bit CPU. A lot of researches on public keys in sensor networks have been conducted in the literature [3, 4, 7]. However, compared with symmetric keys, public keys are still expensive for sensor networks: they take more time to process, and consume more energy. If sensor nodes keep executing public key operations, their energy will quickly get depleted.

Additive Increase Multiplicative Decrease (AIMD) is a frequently used technique to control the traffic of networks. The most noticeable application of AIMD is the congestion control scheme in TCP/IP [10]. The use of AIMD in general networks has been studied extensively, such as Yang and Lam [15], Chiu and Jain [1]. In sensor networks, AIMD has also been used to implement rate control. For example, Rangwala et al. proposed an interference-aware fair rate control protocol in [11], where AIMD control law is used to converge a fair and efficient rate control. and Woo and Culler [13] proposed a rate control mechanism where sensor nodes adjust their transmission rate based on whether the previous packet has been successfully forwarded or not. Wan, Eisenman and Campbell proposed CODA [12], which samples the channel load periodically, and compare the fraction of time that the channel is busy to the optimal channel utilization. There are other schemes that use AIMD technique, but they are based on the assumption that sensor nodes will honestly follow the protocol and refrain from sending more messages, which is not true when there are malicious nodes in the network.

3. SYSTEM MODEL AND DESIGN GOAL

We describe our system model and design goals in this section, as well as the notations used in the description of the scheme.

3.1 Attacking model

In this paper, we assume that the goal of the attackers is to exhaust the energy of the sensor nodes, and to increase the response time of the sensor nodes to the authentic broadcast messages. The primary attacking method of the adversaries is to broadcast large number of faked messages. In order to fool honest nodes, attackers may forward authentic messages from time to time. To implement the attack, adversaries can compromise honest nodes, or deploy malicious sensors of their own. There are other types of DoS attacks such as jamming or black hole attack, but we do not consider them in this paper.

We assume that the attacks are static: adversaries, as well as sensor nodes and base stations, stay in fixed locations throughout the attack. That is, the topology of the network is fixed. Attackers can choose their locations, or take multiple identities, but they cannot move during the attack.

3.2 Design goal

Our goal is to defend sensor networks against DoS attacks, especially the type of attacks that aim at exhausting the energy of sensor nodes. Due to the wireless nature of sensor networks, it is impossible to design a scheme that is totally immune to DoS attacks, so our goal is to reduce the damage of the attacks on the entire network. In other words, we want to contain the damage of DoS attacks to a small portion of the sensor nodes.

Specifically, our design goal includes: (1) *Effectiveness:* the proposed scheme should be effective in containing the damage of DoS attacks to a small portion of sensor nodes; (2) *Efficiency:* the proposed scheme should not bring too much extra cost to the sensor nodes; (3) *Responsiveness:* the proposed scheme should not intro-

duce too much broadcast delay for authentic messages; (4) *Flexibility:* the proposed scheme should be able to adapt to the various needs of different applications.

Notations The following notations are used in the description of the scheme. The explanation of these parameters will be discussed in detail in Section 4.

- m: broadcast message.
- t: unit timeslot.
- ω: current authentication window on sensor nodes.
- ψ_f and ψ_s: updating functions of ω.
- d_a: number of hops m has passed since its last authentication.
- δ: the intensity of attack (i.e., ratio of the number of faked messages and that of the authentic ones).
- k: the number of authentic broadcast messages during unit timeslot.

4. A DYNAMIC WINDOW SCHEME TO CONTAIN DOS ATTACK

To minimize the damage of DoS attacks, sensor nodes need to drop faked messages as early as possible; they need a mechanism to effectively find out where the malicious nodes are, and drop the faked packets from those malicious nodes. The authentication-first scheme can achieve this, but the delay caused by this scheme is not affordable. The ideal solution is, sensor nodes know which nodes are malicious: messages from these nodes are verified before forwarded, while messages from other sources are forwarded before verified. However, this is hard to do: malicious nodes always pretend to be forwarding messages instead of initiating new ones; honest nodes have no idea whether on they are the first-hop victims of malicious nodes or not.

A different angle to look at the problem is: is it possible that sensors gradually shift toward authentication-first mode in a way such that eventually, only the first-hop victims of the attackers stay in authentication-first mode?

4.1 Dynamic Window Scheme

4.1.1 Scheme overview

In the dynamic window scheme, each sensor node s needs to maintain a new parameter: authentication window size (ω). This parameter specifies the largest number of hops an incoming message can be forwarded without being verified. Correspondingly, each broadcast message m keeps record of a new field: distance (d_a), which is used to record the number of hops the message has passed since its last authentication.

When node s receives message m, s compares the authentication window size (ω) with the number of hops m passes since its last authentication (d_a). If $d_a < \omega$, s is in the forwarding-first mode: it increases d_a, and forwards m without verification. However, if $d_a \geq \omega$, s is in the authentication-first mode, which authenticates m first: if the authentication fails, s drops m; otherwise, s resets d_a to 0, and forwards m to its next hop neighbors.

We notice that in broadcast authentication, sensor nodes always authenticate incoming messages. So what really matters in our scheme is when the authentication happens: it can be before the messages are forwarded, or afterwards. In either case, if the authentication fails, s decreases its own ω value; otherwise, s increases ω.

An example is given in Figure 1(a), in which S is the base station, and A, B are sensor nodes 4 and 5-hop away from the base station. At some point, the sizes of the authentication window of A and B are 4. When S broadcasts a new message m, intermediate nodes (shaded ones in the figure) will increase the d_a field of m. At node A, the ω value of A ($\omega = 4$) is compared with the d_a value of m ($d_a = 3$). Since $d_a < \omega$, A is in the forwarding-first mode, which will increase d_a to 4, and forward m without verification. At B, now that $d_a = \omega$, B is in authentication-first mode: it will authenticate m first. If m is authentic, B resets d_a of m to 0, and then forwards it; if m is faked, B will drop m.

4.1.2 Scheme explained

The dynamic window scheme includes the following steps: system initialization, message broadcast, message forwarding and updating, and authentication window size modification. Below is the detailed explanation:

1. System initialization Prior to deployment, the authentication window size of each sensor, $\omega_i, i = 1, \cdots, n$, is initialized as ω_{max}, the largest possible number of hops sensor nodes away from the base station. This means that all sensor nodes are put in *forwarding-first* mode. This is to minimize the initial broadcast delay. Window size updating functions are also loaded into the sensor nodes.

2. Message Broadcast When base station broadcasts a message m, the d_a field of m is set to "0". Base station will then broadcast m to its neighbors, which will relay m to nodes far away from the base station.

3. Message forwarding and updating When node s receives a message m', it will compare the value of its own window size (ω_s) with the d_a field of m'. If $\omega_s > d_a$, then s will increase the d_a value of m', and forward m' without verification. If $\omega_s \leq d_a$, s will check the validity of m' first: if m' is authentic, it will be forwarded, and d_a is reset to 0; otherwise, it will be dropped.

4. Authentication window size updates No matter whether s verifies m' before forwarding it or afterwards, if m' is authentic, ω_s is increased, unless the upper limit of ω_s is reached (ω_{max}); if m' is faked, ω_s is decreased, unless the lower limit of ω_s is reached (ω_{min}). In the future, we use ψ_f to indicate the increasing function, and ψ_s to indicate the decreasing function of ω_s.

Sensor nodes will follow these procedures until every node in the network has a copy of the message (if it is authentic). In the case that the message is faked, it will be dropped by the intermediate nodes. The whole process is illustrated in Figure 1(b).

4.2 Properties of the basic scheme

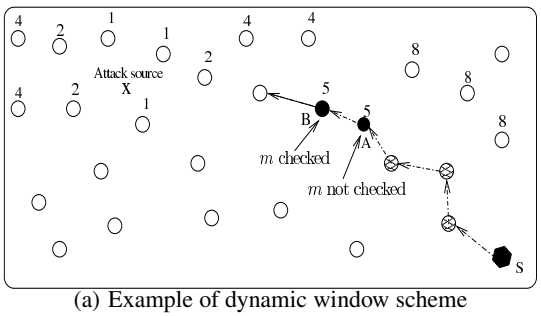
In this section, we discuss the properties of the basic dynamic window scheme, where there is just a single attacker, and the window size updating functions follow basic AIMD law: $\psi_f(\omega) = \omega + 1$ (unless ω_{max} is reached), and $\psi_s(\omega) = \lfloor \frac{\omega}{2} \rfloor$ (unless ω_{min} is reached). We will extend to multi-source attacks and general AIMD functions later in this paper. To simplify discussion, we assume that the attacking ratio is δ: among unit time t, there are k authentic messages, and δk faked ones.

4.2.1 Different patterns of DoS attack

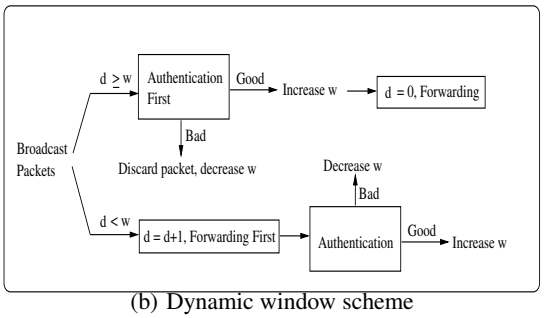
One question we may ask is, from the attacker's point of view, how to maximize the damage of the DoS attacks? Consequently, what impact does it have on sensor nodes?

Before we answer the questions, we notice that for any scheme, so long as the decreasing is faster than the increasing, the final window size will converge to the minimum size allowed. A natural extension to our dynamic window scheme is that so long as the decreasing of authentication window size is faster than the increasing, the authentication window size on sensor nodes will converge to one, which is the minimum value allowed.

In the attacks we study, nodes one hop away from the attacker



(a) Example of dynamic window scheme



(b) Dynamic window scheme

Figure 1: Illustration of dynamic window scheme.

always receive faked messages. For basic AIMD law, the decreasing is faster than the increasing, so if the number of faked messages outweighs that of the authentic ones, the window size of the nodes one hop away from the malicious node will converge to one (ω_{min}) at some point. Therefore, to simplify our analysis, we always choose unit time t such that at the beginning of t, the authentication window of the nodes one hop away from the attacker is one.

When multiple sensor nodes transmit messages at the same time, collision may happen: sensor nodes may need to contend for available channels. For the sake of simplicity, we do not consider message collision in our analysis. In our simulation, however, we will study the effect of packet loss caused by the collision and channel contentions on the dynamic window scheme.

We classify the DoS attacks into three types: Non-consecutive Authentic message Attack *(NAA)*, All-consecutive Authentic message Attack *(AAA)*, and Mixed-Authentic message Attack *(MAA)*. Figure 2 illustrates the three different types of attacks.

Non-consecutive Authentic message Attack *(NAA)* In this type of attacks, there are no consecutive authentic messages. As illustrated in Figure 2(a), between every pair of authentic messages (black dots in the figure), there are always faked messages (hollow dots).

We can prove that, under this scenario, the attack can be easily contained: faked messages will be dropped by the first two hop nodes of the malicious attacker. This is shown in the following property:

Property 1 *If there are no consecutive authentic messages during DoS attacks, faked messages will eventually be dropped by the first two hops of the sensor nodes.*

We provide the sketch of the proof of the property here: at the beginning of the attack, the authentication window of sensor nodes one hop away from the attacker is one (we can always choose unit timeslot this way), which means under this situation, faked messages will be checked and dropped before the possible forwarding by the one-hop nodes. When one authentic message is present, the window size of nodes one hop away is increased to two, which means succeeding faked messages can reach nodes two hops away. In turn, window size for those two-hop nodes will converge to one. Since there are no consecutive authentic messages, after this point, the window size of the two hop nodes will never exceed two, which means every faked message will be verified and dropped by them.

We want to emphasize that the above proof is by no means complete. Rather, it is an outline of one possible way to prove it. We will provide more formal and complete proofs in our more detailed work. But this property does show that our scheme is quite robust against DoS attacks if the attack is intense: if faked messages far outweigh the authentic ones, it is possible that no two authentic messages are consecutive. In this case, little damage would be made to the entire network.

All-consecutive Authentic message Attack *(AAA)* If the attacker wants to affect as many nodes as possible, he needs to arrange the attack such that all authentic messages are transmitted consecutively, which is illustrated in Figure 2(b). As shown in the next property, the faked messages can affect most nodes this way.

Property 2 *Given k authentic messages and δk faked ones, if the k authentic messages are consecutive, faked ones will reach the most sensor nodes. In this case, at least $\delta k - \lceil \log(k+1) \rceil$ faked messages are dropped by the one-hop nodes.*

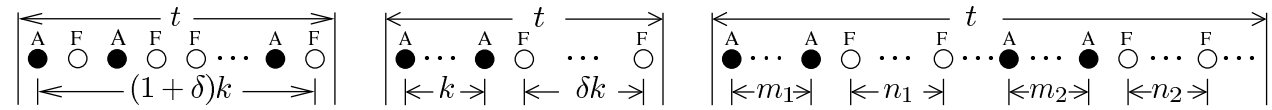
The way to prove this property is similar to the previous one. The basic idea is, in these attacks, the authentication window will reach its maximum value only when the k authentic messages are transmitted consecutively. We skip the complete proof in this paper, but we will provide the formal proof in our more detailed work. This property tells us that, the damage of this type of DoS attacks to our scheme is also contained: $\delta k - \lceil \log(k+1) \rceil$ faked messages will never pass the first hop sensor nodes.

Mixed-Authentic Message attack *(MAA)* *NAA* and *AAA* represent two extreme types of DoS attacks, but the damages of *NAA* and *AAA* are both limited. *AAA*, in particular, is only meaningful theoretically. Smart attackers may implement attacks where there are no such explicit relationships, which is illustrated in Figure 2(c)

This attack is difficult to analyze, but we can partition all the $k + \delta k$ messages based on those authentic messages. For example, we assume that in a smaller timeslot t', the authentication window size of a node close to the malicious node is 1, and there are m_1 consecutive authentic messages, followed by n_1 faked messages, and then m_2 authentic messages followed by n_2 faked messages. We can view the smaller unit timeslot t' as a special case of *AAA*. Then, so long as $n_2 \geq \log\lceil m_2 + \log_{2^{n_1}}(1+m_1) \rceil$, the window size of node becomes 1 during n_2. We will provide more detailed discussions in our future works.

This can be extended to the more general cases: if the distribution of the messages is $m_1, n_1, \cdots, m_i, n_i$, where m_i refers to authentic messages, and n_i refers to faked messages, we can always treat the smaller timeslots as special cases of AAA. We will provide more analysis and discussion of this in our future work.

For nodes one hop away from the attacker, they will always receive δk faked messages. Besides the number of faked message they receive, what matters most is the number of faked messages they forward. Since if the authentication window of nodes one hop away from the attacker is one, the succeeding faked messages will



(a) Non-consecutive Authentic message Attack (*NAA*) (b) All-consecutive Authentic message Attack (*AAA*) (c) Mixed-Authentic message Attack (*MAA*)

Figure 2: Patterns of DoS attacks.

be dropped, we can use the formulae obtained earlier to estimate the number of dropped packets in *MAA*. They are not in closed form, but they can serve as a criteria to evaluate the performance of the scheme.

4.2.2 Energy saving in the presence of DoS attacks

Reducing the amount of faked messages received by the sensor nodes is of vital importance for DoS resistant schemes. This is because, the energy saving for the sensor nodes is comprised of two parts: energy saving on communication (receiving/forwarding packets), and energy saving on computation (signature authentication). In our analysis, we do not calculate exactly how much *Joule* of energy is saved. Rather, we focus on the percentage of faked messages the sensor nodes receive, and the percentage of nodes that are affected by the faked messages.

THEOREM 4.1. *In NAA, sensor nodes two hops away from the attacker are immune from the attack; in AAA, sensor nodes more than two hops away from the malicious attacker will receive at most* $\lceil \log(k+1) \rceil$ *faked messages.*

Proof: The correctness of this Theorem can be directly obtained from Property 1 and 2. From Property 1, we know that faked messages will be dropped by the nodes two hops away from the malicious node; from Property 2, we know that at most $\lceil \log(k+1) \rceil$ faked messages can reach nodes more than two hops away from the malicious node. These are exactly the conclusion of the Theorem.

We can further study the overall energy savings on sensor networks for all the nodes. For example, assume the density of the network is d, the transmission range of sensor node is r, and the total number of sensor nodes is n. Then, for *NAA*, only rd nodes will be affected by the faked messages, which means, $(n - rd)$ nodes will not waste energy on the δk faked messages. The overall energy saving will be at least $(n - rd)\delta k$. Similarly, for *AAA*, the lower bound of overall energy saving is $(n - rd)\lceil \delta k - \log(k+1) \rceil$.

For *MAA*, it is difficult to obtain a closed form of energy savings, but we can use the formula obtained in the discussion of *MAA* to estimate the energy savings: if the authentication window of sensor nodes one hop away from the malicious nodes becomes one, succeeding faked messages are dropped, which means that the rest of nodes save energy on those faked messages. We must emphasize that this is only the lower bound of energy savings for the nodes two hops away from the malicious node. In most cases, the energy saving is much larger.

4.2.3 Broadcast delay for authentic messages

Broadcast delay in our scheme is determined by the number of intermediate nodes that verify the incoming message before forwarding it. To calculate the broadcast delay of our scheme, we need to find out how many intermediate nodes are in the authentication-first mode.

Assume that for a node i-hop away from the base station, the intermediate nodes are $s_1, s_2, \cdots, s_i$. Correspondingly, the authentication window sizes on those nodes are $\omega_1, \omega_2, \cdots, \omega_i$. If we use $v_j, 1 \leq j \leq \omega_{max}$ to indicate the number of nodes whose window size is j, then we observe the following interesting property:

Observation *If* $v_1, v_2, \cdots, v_i$ *are sorted (in increasing or decreasing order), then the number of nodes that are in authentication-first mode is at most* $\sum_{j=1}^{j=\omega_{max}} \lceil \frac{v_j}{j} \rceil$.

Again, we omit the proof of this observation in this paper, but will provide a formal proof in our more detailed work. The observation provides a way to estimate the upper bound of broadcast delay for a message to reach nodes i-hop away from the base station. In the dynamic window scheme, however, it is possible that $v_1, \cdots, v_i$ are not sorted. In that case, we can always divide the unsorted array of $v_1, \cdots, v_i$ to smaller arrays where they are sorted. Assume ns is the number of such smaller sorted sub-arrays, v_{hj} refers to the number of nodes whose window size is j in sub-array h, and t_v is the time to authenticate a broadcast message, then we can calculate the upper limit of broadcast delay using the following formula: $t_{delay} \leq t_v \cdot \sum_{h=1}^{h=ns} \sum_{j=1}^{j=\omega_{max}} \lceil \frac{v_{hj}}{j} \rceil$. Again, this formula is not in a closed form, but it can be used to analyze the broadcast delay of authentic messages to nodes i-hop away from the base station.

4.3 Extension of the Basic Scheme

The window size updating functions play a very important role in our scheme, so what should we expect for those functions? Moreover, previous discussions are based on single attacker scenario, but in reality, there may be multiple attackers, then how will these multiple attackers affect the scheme? We will discuss these issues in this section.

4.3.1 Window size updating functions

Requirements for window size updating functions The decreasing and increasing of the authentication window size are important. For the scheme to be effective in containing DoS attacks, window size updating functions should have the following properties: (1) *Gradient distribution*: sensor nodes close to the attacking source should have smaller windows than the nodes far away from the attacker; (2) *Fast decrease*: upon a failed authentication, the authentication window should be decreased rapidly so that the network can quickly contain DoS attacks; (3) *Slow increase*: upon a successful authentication, the window should be increased slowly; otherwise, attackers can take advantage of this by mixing faked messages with authentic ones, thus easily defeat the containment.

AIMD technique that is used in the basic scheme is quite efficient, which will not introduce too much computing overhead to sensor nodes. We want to know more about the window size updating functions, which is discussed as the following.

General window size updating functions In the discussion of the basic scheme, we use basic AIMD law: $\psi_f(\omega) = \omega + 1$, and $\psi_s(\omega) = \lfloor \frac{\omega}{2} \rfloor$, $\omega_{min} \leq \omega \leq \omega_{max}$. Other increasing and decreasing functions may be used to improve the performance of the scheme. When applying general AIMD laws to our study, we can assume that the increasing function is $\psi_f(\omega) = \omega + \alpha$, and the de-

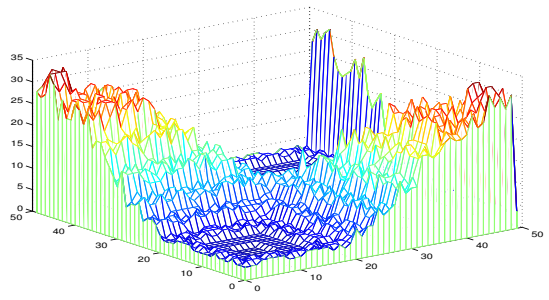


Figure 3: Window size on sensors under multi- attacks.

creasing function is $\psi_s(\omega) = \omega/\beta$, where $\alpha > 0, \beta > 1, \omega_{min} \leq \omega \leq \omega_{max}$. So long as the decreasing of the authentication window is faster than the increasing, Property 1 in Section 4.2.1 still holds, but Property 2 needs to be modified.

Property 2.1 *(Extension of Property 2) Given k authentic messages and δk faked messages, and given $\psi_f(\omega) = \omega + \alpha$, $\psi_s(\omega) = \omega/\beta$, if the adversary tries to affect most nodes, at least $\{\delta k - \lceil \log_\beta(\alpha k + 1) \rceil\}$ faked messages are dropped by the first hop of the malicious nodes.*

Again, the proof of the property is similar to that of Property 2, and we leave the complete proof in our more detailed work. What we know from these properties is that, the energy saving depends on both the increasing and decreasing functions. We can choose the appropriate functions that fit our needs. For example, one intuitive observation is that, the faster the authentication window increases, the smaller number of nodes in the authentication-first mode there will be, thus the smaller broadcast delay. However, in that case, we need to carefully choose the decreasing functions. Moreover, it is possible that we use non-linear increasing or decreasing functions. That will be more complicated to analyze though. We will further study the impact of the various ways to update the window size on sensor nodes in section 5.

4.3.2 Multi-source attacks

In hostile environments, the adversaries can compromise or inject multiple nodes into the network to implement DoS attacks. The damage to the network will be more severe, because now sensor nodes may receive faked messages from multiple sources.

However, the dynamic window scheme can handle these multiple attackers, and the damage caused by the attackers is still limited to only a portion of the sensor nodes. Figure 3 illustrates this. In this example, two malicious nodes located at $(10, 10)$ and $(35, 35)$ keep broadcasting faked messages. Sensor nodes close to them will be affected, but for nodes far away from both of them, the impact is quite limited: faked messages broadcasted from the malicious nodes are dropped by the intermediate nodes. Generally speaking, the multiple malicious nodes will divide the entire network into several smaller sub-areas, and the sensor nodes close to the attackers will have smaller authentication window than nodes far away. When a message arrives at these nodes, it is more likely that this message will be verified before being forwarded.

Multiple attackers will affect the broadcast delay in our scheme. When the window size on sensor nodes becomes smaller, nodes are more likely to be in the authentication-first mode, and the incoming messages are more likely to be authenticated before forwarded. We can still use the formula derived in Section 4.2.3 to estimate the upper bound of broadcast delay. The only problem is, the delay will be larger than that in the single source attack, since the authentication window of the sensor nodes tend to be smaller. In other words, $\sum_{h=1}^{h=ns} \sum_{j=1}^{j=\omega_{max}} \lceil \frac{v_{hj}}{j} \rceil$ becomes larger. Since this is the upper bound of the delay, the situation may not be too bad, as messages may take alternative routes to their destinations. We will further investigate the issues in this area in our future research.

5. EVALUATION AND ANALYSIS

The purpose of DoS attacks can be multi-folded: exhaust the energy of sensor nodes, prevent sensor nodes from receiving authentic messages, or increase the response time sensor nodes receive messages. The proposed dynamic window scheme can limit the damage of DoS attacks to a portion of sensor nodes, but some parameters may have significant impacts on the performance of the scheme. such as the window size on each sensor node, the intensity of the DoS attacks, the number of one-hop neighbors of the sensor nodes, etc.

In this section, we study the effect of various parameters on the performance of the proposed scheme by comparing the performance of our scheme with that of the forwarding-first scheme and the authentication-first scheme. The criteria of our evaluation are the energy savings of all the sensor nodes, and the delay for authentic messages to reach sensor nodes far away. To be more specific, we evaluate the following metrics: (1) *Average delay of authentic broadcast message*, which measures how long it takes for each sensor to receive a legitimate packet; (2) *Portion of nodes that receive faked messages*, which shows how much energy is wasted on receiving and verifying those faked messages; (3) *Portion of nodes that forward faked messages*, which indicates how effective the dynamic window scheme is in containing DoS attacks.

5.1 Environmental setup

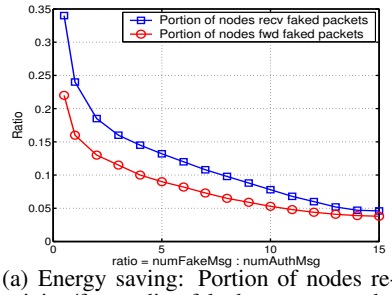
In our simulation, 5000 sensor nodes are randomly deployed into an area of 200m×200m, with the transmission range of sensor nodes set as 6m. We assume that it will take 2 seconds for a node to authenticate a message (signature authentication). As discussed earlier, we assume that base stations, as well as attackers, are located at the fixed locations. We simulate the Mixed-Authentic Message Attacks, as this is more realistic in the real applications. We assume that the malicious nodes keep sending faked messages, but they may also forward authentic messages from time to time. Initially, the authentication window size on each sensor node is 64 (ω_{max}).

Unless specified otherwise, we assume single source attack, and the window size updating functions follow basic AIMD law: $\psi_f(\omega) = \omega + 1, \psi_s(\omega) = \lfloor \omega/2 \rfloor, 1 \leq \omega \leq \omega_{max}$. In the experiments that we design, these parameters (or functions) may change in order to evaluate the performance of the dynamic window scheme.

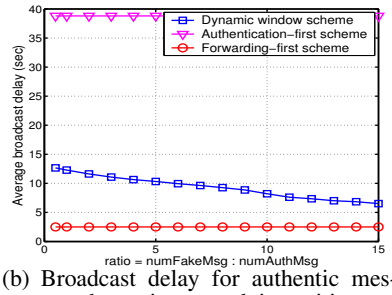
5.2 Simulations and results

Intensity of DoS attacks: We simulate the attacking scenarios in which the ratio between the number of faked messages and authentic messages ranges between 0.5 and 15. This means, the faked messages make up 33% to 94% of the total messages. The results are shown in Figure 4.

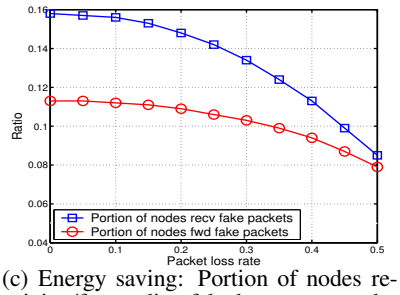
Figure 4(a) shows that only a small portion of the nodes will receive faked messages, with an even smaller portion of the nodes forwarding the faked messages. More importantly, when the attacks become more intense, the scheme performs even better. The reason is that when the attacks become more severe, the dynamic window scheme can isolate the malicious nodes more quickly. If there are fewer faked messages than authentic messages, the performance of our scheme, specifically, energy saving, may not be very impressive. It is still good though, as illustrated by the first cou-



(a) Energy saving: Portion of nodes receiving/forwarding faked messages under attacks of various intensities.

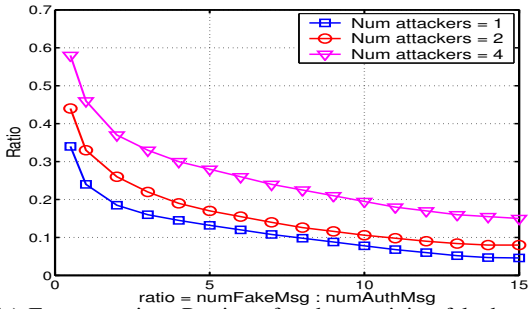


(b) Broadcast delay for authentic messages under various attack intensities

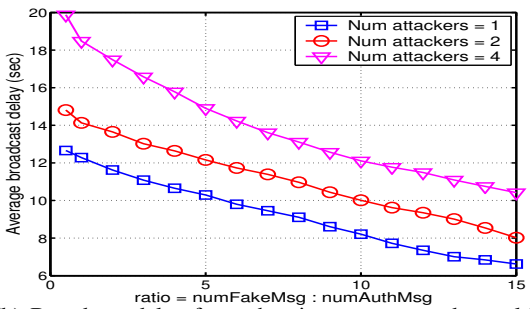


(c) Energy saving: Portion of nodes receiving/forwarding faked messages under various packet loss rates

Figure 4: Effects of various DoS attack intensities and packet loss rates.



(a) Energy saving: Portion of nodes receiving faked msg under multiple attackers.



(b) Broadcast delay for authentic messages under multiple attackers.

Figure 5: Effects of multiple attackers.

ple of markers in Figure 4(a). However, for DoS attacks, usually faked messages far outweigh the authentic ones. In that case, our scheme can filter out most of the faked messages, and thus achieve more significant energy saving. This is a clear indication that the proposed scheme is robust against DoS attacks.

We also notice that the proposed scheme does not introduce too much broadcast delay. As shown in Figure 4(b), in the presence of faked messages, the delay for authentic broadcast message remains relatively small: on average, the delay of the dynamic window scheme is about one-fourth of that of the authentication-first scheme.

Packet loss rate: Packet loss can be caused by many reasons. For example, when multiple sensor nodes transmit messages simultaneously, collision may happen, and some packets get lost. This is a serious issue for broadcast authentication. In the experiments conducted in this part, we want to study the impact of packet loss on our scheme. To make the comparison, we change the packet loss rate between 0 and 0.5 in our simulation. The results are show in Figure 4(c).

It is interesting to observe that the dynamic window scheme achieves good energy saving when there are packet losses. Especially, when the packet loss rate increases, the number of nodes receiving faked messages decreases exponentially. We do notice that packet loss itself may contribute to the energy saving: faked messages never reach some intended receivers due to the packet loss. However, this experiment does show that the proposed scheme can achieve good energy savings when there are packet losses.

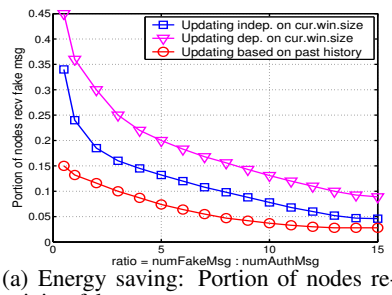
Multiple malicious attackers: In this experiment, we want to study the effect of multiple attackers on our scheme. In our simulation, there is one base station, but there are multiple attackers in the network. To keep the same attacking intensity as the single attacker case, each malicious attacker will send out a portion of the faked messages. Figure 5 shows the result.

In these figures, we can see that, when there are multiple attackers in the network, the performance of the dynamic window scheme deteriorates: more sensor nodes will receive faked messages, and longer times are needed for authentic messages to reach nodes far away from the base station. We do notice that our scheme can still filter out most of the faked messages, while the delay is not too bad. Moreover, the cost for the adversaries to implement the attacks is dramatically increased.

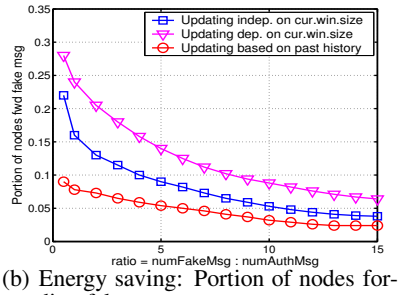
Window size updating functions: Experiments in this section are used to study the impact of various window size updating functions. We study the effect by comparing three different approaches to update the authentication window of sensor nodes: (1) updating is independent of the current window size; (2) updating depends on current window size; and (3) updating is based on the validity history of the incoming messages.

Specifically, in approach (1), we adopt basic AIMD law: $\psi_f(\omega) = \omega+1, \psi_s(\omega) = \lfloor \omega/2 \rfloor$; in approach (2), we use an improved AIMD law: $\psi_f(\omega) = \omega + \lfloor \omega/4 \rfloor, \psi_s(\omega) = \lfloor \omega/2 \rfloor$; and for approach (3), we collect the last 10 messages that the sensor nodes receive: assume there are α authentic messages in these 10 messages, then $\psi_f(\omega) = \omega + \lfloor \alpha\omega/10 \rfloor, \psi_s(\omega) = \lfloor (10-\alpha)\omega/10 + 1 \rfloor$. In all the above cases, $1 \leq \omega, \psi_f(\omega), \psi_s(\omega) \leq \omega_{max}$.

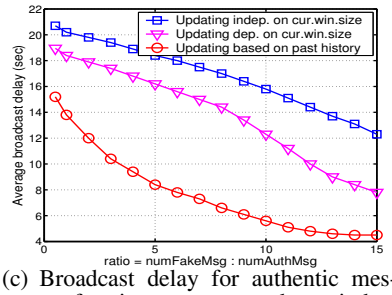
The results of the experiments are shown in Figure 6. As shown in these figures, the third approach, updating window size based on the validity history of the incoming messages, is the best in terms of containing DoS attacks (smallest number of nodes receiving and forwarding faked messages), and broadcast delay (lowest broad-



(a) Energy saving: Portion of nodes receiving fake messages.

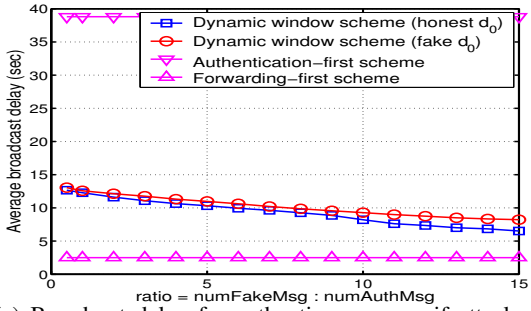


(b) Energy saving: Portion of nodes forwarding fake messages.

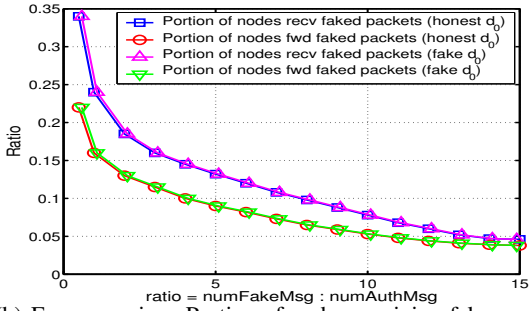


(c) Broadcast delay for authentic messages of various ways to update window size.

Figure 6: Effects of various ways to update window size.



(a) Broadcast delay for authentic messages if attackers change d_a.



(b) Energy saving: Portion of nodes receiving fake messages if attackers change d_a.

Figure 7: Handling attacks that change d_a value.

cast delay). However, its cost is also the largest among the three, because sensor nodes need to remember more information. These figures also show interesting relationship between the first and the second approach: the broadcast delay for the first approach is larger than that of the second one, but the energy saving of the first approach is better. This means, there is a tradeoff between these two properties. We can adjust the parameters of our scheme to fit with the specific needs of various applications.

Handling faked d_a value: Messages (authentic or faked) forwarded by the malicious nodes may contain a bogus distance value (d_a). The malicious nodes can assign a random initial d_a value to a faked message, or change the d_a field of the honest messages. In this part, we conduct experiments to study what would happen if the malicious attacker does change the d_a value. The result is show in Figure 7.

In our tests, if the malicious node forwards a faked message, the d_a field of the message is set as 0, but if it decides to forward an authentic message, the d_a field of the message is a random number (instead of increasing by one). This kind of behaviors may cause the most damage to the honest sensor nodes. But as shown in Figure 7(a), the attacker gained little advantage (if any) in increasing broadcast delay, and almost no advantage on exhausting the energy of sensor nodes, as shown in Figure 7(b). The reason is, some honest nodes will check the validity of the messages before they forward those messages, and the number of such nodes is determined by both the broadcast message and the nodes themselves. Again, this is a clear indication that our scheme is robust against DoS attacks.

6. CONCLUSION AND FUTURE WORK

Denial of Service attacks are very difficult to prevent in sensor networks. In this paper, we discussed a specific type of DoS attacks, and classify the different attacking patterns. We also presented a dynamic window scheme that can effectively contain the damage of DoS attacks to a small portion of the nodes. Our scheme allows each individual node to make its own decision on whether to forward a message first or verify it first. Even though sensors have no idea where the malicious attackers are, they can effectively locate the attackers and contain the damage caused by them. Our scheme is efficient, and does not introduce too much broadcast delay. It is also very flexible: the parameters of the scheme can be configured such that the different needs of the various applications are met.

In order to fully evaluate the performance of our scheme, we need to further study the distribution of the size of authentication window on sensor nodes. Also, the window size updating functions are of significant importance in our scheme: experiments show that past history will be especially helpful in improving the performance of our scheme. In our future work, we will keep investigating the possibility to include this in the defending of DoS attacks in sensor network. We will further investigate the impact of multiple attackers in the sensor networks, and consider the per-source or source-class algorithms in defending DoS attacks in sensor networks in the future.

7. REFERENCES

[1] D. Chiu and R. Jain. Analysis of the increase/decrease algorithms for congestion avoidance in computer networks. *Journal of Computer Networks and ISDN*, 17(1):1–14, June

1989.

[2] J. Deng, R. Han, and S. Mishra. Defending against path-based dos attacks in wireless sensor networks. In *ACM SASN '05*, Alexandria, VA, USA, November 2005.

[3] H. Eberle, S. Shantz, V. Gupta, N. Gura, L. Rarick, and L. Spracklen. Accelerating next-generation public-key cryptosystems on general-purpose cpus. *IEEE Micro*, 25, March 2005.

[4] G. Gaubatz, J. Kaps, and B. Sunar. Public keys cryptography in sensor networks – revisited. In *Proceedings of ESAS 2004*, 2004.

[5] N. Gura, A. Patel, A. Wander, H. Eberle, and S. Shantz. Comparing Elliptic Curve Cryptography and RSA on 8-bit CPUs. In *CHES 2004*, Cambridge, MA, August 11-13 2004.

[6] J. McCune, E. Shi, A. Perrig, and M. Reiter. Detection of denial-of-message attacks on sensor network broadcasts. In *Proceedings of the IEEE S&P*, Oakland, CA, USA, May 2005.

[7] P. Ning and A. Liu. Tinyecc: Elliptic curve cryptography for sensor networks. Cyber Defense Laboratory in NCSU, September 2005.

[8] A. Perrig, R. Canetti, D. Song, and D. Tygar. Efficient and secure source authentication for multicast. In *Proceedings of NDSS*, San Diego, CA, February 2001.

[9] A. Perrig, R. Szewczyk, V. Wen, D. Cullar, and J. D. Tygar. Spins: Security protocols for sensor networks. In *Proceedings of MobiCom*, pages 189–199, Rome, Italy, July 2001.

[10] DARPA INTERNET PROGRAM. Rfc 793 - transmission control protocol, September 1981.

[11] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis. Interference-aware fair rate control in wireless sensor networks. In *In Proceedings of SIGCOMM*, Pisa, Italy, September 2006.

[12] C. Wan, S. Eisenman, and A. Campbell. Coda: congestion detection and avoidance in sensor networks. In *Proceedings of Sensys*, Los Angeles, LA, USA, November 2003.

[13] A. Woo and D. Culler. A transmission control scheme for media access in sensor networks. In *Proceedings of ACM MobiCom*, Rome, Italy, 2001.

[14] A. D. Wood and J. A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62, 2002.

[15] Y. Yang and S. Lam. General aimd congestion control. In *Proceedings of ICNP*, Osaka, Japan, November 2000.