Understanding Dynamic Denial of Service Attacks in Mobile Ad Hoc Networks

Fei Xing Wenye Wang
Department of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC 27695, USA
fxing@ncsu.edu, wwang@ncsu.edu

***Abstract*— In mobile ad hoc networks (MANETs), Denial of Service (DoS) attacks not only consume the scarce system resources, such as bandwidth, battery energy, or CPU cycles, but also isolate legitimate users from a network. Therefore, DoS attacks may impact the network connectivity seriously and may further undermine the networking functions, such as control and data message delivery. In this paper, we will present a deep insight into DoS attacks and their impacts on MANETs. First, we analyze the node isolation problem resulting from DoS attacks and derive the probability of node isolation, which shows that the DoS attack exploiting fraudulent routing messages, such as *BlackHole* attack, impacts the connectivity much severer than other attacks. Second, we notice that the node mobility and potential attack propagation have hardly been considered in the previous DoS attack studies; therefore, we introduce a *dynamic DoS attack* in this paper. The dynamic DoS attack is characterized in exploiting the node mobility, dynamic power control, and compromised nodes to spread new DoS attacks dynamically. Further, we provide an analytical study on the properties of this new DoS attack, and explain its potential devastating impact on the connectivity of MANETs.**

I. Introduction

Since mobile ad hoc networks (MANETs) do not require pre-existing infrastructures and can be deployed spontaneously, MANETs are suited to a plenty of applications both in civilian environments, such as disaster relief and spontaneous conference, and in military environments, such as battlefield deployments and sensor networks. Originally, MANETs are expected to be more reliable than structured networks because all network entities should work cooperatively and the notorious single-point-failure problem may be avoided by distributed computing. However, compared with the wired networks, MANETs are more vulnerable to security attacks due to their unique features, such as stringent power constraints, error-prone communication media and highly dynamic network topology. *Confidentiality, integrity* and *availability* are three major requirements of the information security for any system. To achieve confidentiality and identity, cryptographic solutions used in wired networks can be used in MANETs as well. However, the availability of MANETs has been challenged by Denial of Service (DoS) attacks because DoS attacks may impact the network connectivity seriously and further undermine the networking functions, such as control and data message delivery.

According to the layered network reference model, MANETs are vulnerable to the DoS attacks on the *link layer* and the *network layer*. A DoS attack is said to be on the link layer when it can be launched by exploiting any vulnerabilities of data link layer protocols. For example, an attacker may use the binary exponential back-off scheme of IEEE 802.11 to deny access to the wireless channel from its local neighbors [1]–[3]. Correspondingly, DoS attacks on the network layer take the advantage of the vulnerabilities of the network layer protocol, which can be further classified into three types, i.e., routing disruption, forwarding disruption, and resource consumption attacks. For example, Wormhole (Rushing) [4], [5] and BlackHole attack [6] are routing disruption attacks, and JellyFish, directional antenna abusing [6], and dynamic power abusing attacks [6], [7] are forwarding disruption attacks, while Packet injection attacks [8] and control packet floods [9] are resource consumption attacks.

Although many efforts (including above) have been done on the impact of DoS attacks in MANETs, few of them analyzed the impact on the connectivity, which is an essential requirement for any networks, especially military networks. We also notice that all existing DoS attacks studied are *static* because the mobility of misbehaving nodes is ignored and the potential propagation of DoS attacks is un-investigated. Therefore, in this paper, we first give a comprehensive overview to the existing DoS attacks, both on link layer and on network layer, in MANETs. Based on the review of known DoS attacks, we then provide a detailed analysis on the node isolation problem resulting from these DoS attacks to reveal the impact of DoS attacks on the network connectivity and derive the probability of node isolation. Considering the node mobility and attack propagation, we next introduce a new DoS attack called *dynamic* DoS attack by using numerous examples, which illustrate how a malicious node can enlarge the effective scope of DoS attacks and how DoS attacks may propagate by compromising cooperative neighbors. We also model the dynamic DoS attack propagation by a simple semi-Markov process to evaluate the propagation rate of DoS attacks. The analytic results will show that the dynamic DoS attack armed with propagation ability can harm the network connectivity more severely and quickly.

The remainder of this paper is organized as follows. In Section II, we introduce the network and security assumptions used in this paper. In Section III, we give an overview to the existing DoS attacks and analyze the node isolation problem in MANETs. In Section IV, we introduce the *dynamic DoS* *attack* by examples and analyze its impact on MANETs, in particularly, its propagation speed, followed by conclusions in Section V.

II. Assumptions

A. Network Assumptions

In MANETs, many events may prevent cooperative nodes from accessing network services. For example, nodes may become failed due to software bugs or battery depletion, so a node may be unable to communicate with other nodes if its neighbors are all failed. Nodes may also behave selfishly by not forwarding packets for other nodes in order to save their battery energy, which will also tamper the normal communication service. Nevertheless, the denial of service caused by failures or selfishness is not necessarily intended by failed or selfish nodes. However, once a cooperative node is compromised by malicious nodes, it may become a malicious node and launch aggressive DoS attacks to other cooperative nodes. Because malicious nodes launch DoS attacks explicitly and intendedly, in this paper, we focus on the DoS attacks caused by malicious nodes and their effects.

This work assumes that an ad hoc network comprises a group of mobile nodes communicating through a common broadcast channel using bidirectional communication. The underlying topology of an ad hoc network can be presented by an undirected graph $G = G(V, E)$, where V and E are the set of vertices (nodes) and edges (links), respectively. Two nodes have a link when they are within the transmission range of each other; however, we do not assume that the transmission range is identical for all nodes in that malicious nodes may change their transmission range to launch DoS attacks by exploit dynamic power management techniques. Although the *promiscuous mode* helps some ad hoc routing protocols, such as the Ad hoc On Demand Distance Vector (AODV) routing protocol, to determine link breakages faster, and is expected to be a basic method to detect node misbehaviors [10], [11], we do not assume that the promiscuous mode is used by mobile nodes by default in that it is not appropriate to protect data integrity for all military scenarios.

B. Security Assumptions

Since we focus on the threats to the availability of MANETs, i.e., DoS attacks, in this paper, other security issues, such as message privacy and node identity, are not discussed here. Nevertheless, it is possible that the integrity of all transmitted information, including control messages, data packets and their ACKs, can be protected by using pairwise shared secret keys, digital signatures, or symmetric key protocol TESLA [12]. With the protection of data integrity, attackers can hardly disrupt normal network operations by simply modifying packets from other nodes or impersonating other nodes. However, it is still possible for an interior malicious node to launch DoS attacks against other nodes if it has already possessed legitimate pre-shared keys or signatures.

III. Static DoS Attacks and Impacts

A. Overview of DoS Attacks

A DoS attack is an event that diminishes or eliminates a network's capacity to perform its expected function. Although hardware failures, software bugs, resource exhaustions, environmental conditions, or any complicated interactions between these factors can cause a DoS [13], we consider primarily DoS attacks launched by malicious nodes in this paper. A malicious node can launch DoS attacks on either the link layer or the network layer, which are summarized as follows.

1) DoS Attacks on the Link Layer: IEEE 802.11 medium access control (MAC) protocol is current used as the link layer protocol for MANETs. It was identified that IEEE 802.11 MAC is vulnerable to DoS attacks which exploit its binary exponential back-off scheme [1], [14]. Because a successful transmission leads to a smaller contention window, a continuously transmitting node can always capture the channel and cause other nodes to back off endlessly. A modified back-off scheme was proposed in [2] to solve this attack by providing the back-off timer from the receiver end.

Further, it was noticed that the NAV (network allocation vector) field in the RTS/CTS (request to send/clear to send) frames exposes another vulnerability to DoS attacks. Since a malicious node is aware of the duration of the ongoing transmission in its neighborhood, it can transmit just a few bits to interfere the ongoing link-layer frames with a trivial energy cost. Reference [15] studied this jamming problem in detail and concluded that the Low Density Parity Codes (LDPC) should be used for binary modulation to mitigate the DoS attack.

2) DoS Attacks on the Network Layer: DoS attacks on network layer generally fall into three categories: *resource deprivation, routing disruption*, and *forwarding rejection*.

In a resource deprivation attack, malicious nodes can inject extra control or data packets into the network. For example, if AODV is used for a MANET, a malicious node may keep sending different RREQ messages to its neighbors. Since the sequence numbers or fake destination addresses can be changed each time, an attacker's neighbors are not able to discern if these messages are fake ones or new requests, such that they have to forward to their neighbors and so forth. If the malicious node sends these fake messages at a high rate, its neighbors have to spend much resources, such as bandwidth, CPU cycles, and battery energy, to handle these fake messages. A slightly less aggressive version of this attack was presented in [9] where a malicious node keeps initiating route discovery requests at a lower rate but ignores any reply to them. The simulation results in [9] shown that this malicious control packet flooding attack degrades the network performance. Besides the control packet flooding attack, a malicious node can also inject a large number of junk data packets into the route to consume the resource of intermediate routing nodes. Reference [8] studied this attack and proposed an on-demand and hop-by-hop source authentication protocol in forwarding packets, so called *SAF*, to mitigate this attack.

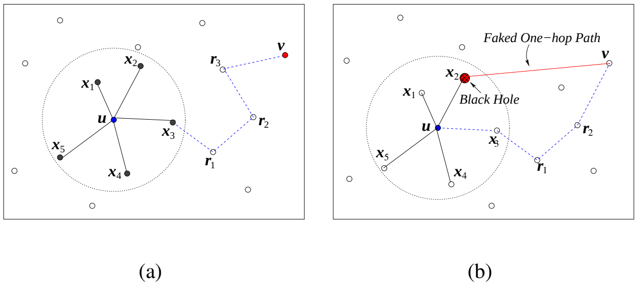
In a route disruption attack, malicious nodes may send forged routing packets to mislead the route selection. A typical example of this attack is *BlackHole* [6], [12], [16], in which an attacker launching the BlackHole attack could route all packets for some destination to itself and then discard them. Another type of routing disruption attack is so called *wormhole*, studied in [4]. To launch the wormhole attack, two malicious nodes M_1 and M_2 are needed to collaborate via a private network connection, e.g., Ethernet cable, such that M_1 can forward the packets received from other nodes directly to M_2 through the wormhole and M_2 can rebroadcast the forwarded packets to another area of the network. A scheme called *packet leashes* was proposed to defend against the wormhole in [4]. The *Rushing attack* was introduced in [5]. To launch the rushing attack, a malicious node disseminates RREPs faster than other nodes. When cooperative nodes receive the later arrived RREPs, they will treat these legitimate RREQs as the duplicates and drop them. Reference [5] also presented a rushing attack prevention protocol (RAP) to thwart this attack.

Compared with the resource deprivation and route disruption attack, malicious nodes launching forwarding rejection attacks may comply with all routing procedures. The *JellyFish* attack was introduced and studied thoroughly in [6]. A malicious node launching JellyFish attacks may keep active in both route discovering and packet forwarding in order to prevent it from detection and diagnosis, but the malicious node can attack the traffic via itself by reordering packets, dropping packets periodically, or increasing jitters. The JellyFish attack is especially harmful to TCP traffic in that cooperative nodes can hardly differentiate these attacks from the network congestion. Reference [6] also described that malicious nodes may even abuse directional antenna and dynamic power techniques to avoid upstream nodes to detect their misbehaviors of dropping packets. A concept of *self-healing community* in [11] was claimed to be able to mitigate the directional and dynamic power transmission attack, which requires the network interface stay in the promiscuous mode; however, in military ad hoc networks, setting network interface cards as the promiscuous mode enables nodes to become sniffers or eavesdroppers, which may lead to other potential security threats.

B. Node Isolation Problem

The connectivity of an ad hoc network is the prerequisite of all multi-hop network operations. When a node has no neighbors, the node is said to be isolated from an ad hoc network. Obviously, the problem of node isolation resulting from lacking active neighbors is a direct reason for network partitioning. In MANETs, malicious nodes can launch DoS attacks, which can isolate a node even if the isolated node has active neighbors. In this section, we will reveal the impact of DoS attacks on the connectivity of MANETs by providing an in-depth analysis on the node isolation problem. In the following context, we use the *BlackHole* and *JellyFish* attack as the representatives of routing disruption attacks and forwarding rejection attacks, respectively.

Case I: In the scenario shown in Fig. 1(a), some of neighboring nodes of node u are failed or selfish nodes. It is clear that if all neighbors of node u are selfish or failed, there is no way for node u to establish communications with other nodes at a distance of more than one-hop away. In this case, we say node u is *isolated* by its selfish or failed neighbors.



(a)

(b)

Fig. 1. Node Isolated by Misbehaving Neighborhood.

Case II: As shown in Fig. 1(b), one of the neighbors of node u is a malicious node, e.g., x_2 is a *BlackHole*. When AODV is used as the routing protocol, node u discovers the route to node v by $RREQ$ messages. Then node x_2 may send a fake $RREP$ message claiming that itself is only one-hop away from node v. In consequence, node x_2 is treated by node u as its next hop, and the *BlackHole* x_2 just drops all packets from node u. In fact, only one *BlackHole* neighbor x_2 is sufficient to trap all traffic initiated from node u if the destination is beyond node u's one-hop neighborhood. Furthermore, a *BlackHole* node, such as x_2, is able to trap all traffics of its neighbors, which implies that a *BlackHole* node may *isolate* all its neighbors.

Case III: A *JellyFish* node may reorder, delay or drop partial packets expected to be forwarded, which may cause packet loss in turn. Let us take an example in Fig. 1(b) where TCP is used as the transport layer protocol. Suppose that node x_2 is a *JellyFish*, and node u starts to communicate with node v after a path via the *JellyFish* node is established. Then the DoS attacks launched by node x_2 will cause packet loss and break off the communications between nodes u and v eventually. Nevertheless, it is still possible for node u to communicate with other nodes if the *JellyFish* neighbor is not the next hop.

From the analysis above, we know that DoS attacks can cause node isolation problem. Generally, if every neighbor of a node is either selfish, malicious, or failed, the node is isolated; further, as long as a node has one *BlackHole* neighbor, the node is also isolated. Therefore, DoS attacks have severe impact on the connectivity of ad hoc networks. We will derive the probability that a node is isolated under DoS attacks.

C. Probability of Node Isolation

From the discussion in the previous section, we know that when all of the neighbors of a node u are either selfish or failed, or if the node u has at least one *BlackHole* neighbor, then the node u will be isolated. This implies that the node isolation probability can be derived with regard to the number of different neighbors.

Let $\mathcal{S} \triangleq \{C(cooperative), S(selfish), M(malicious), F(failed)\}$ be a set of four types of mobile nodes, we introduce the following notations:

$D(u)$	the number of all neighbors of node u
$\hat{n}_i(u)$	the number of u's neighbors in type i, $i \in \mathcal{S}$
$\hat{n}_{BH}(u)$	the number of node u's *BlackHole* neighbors
$\hat{n}_{JF}(u)$	the number of node u's *JellyFish* neighbors

Here we have $\hat{n}_m(u) = \hat{n}_{BH}(u) + \hat{n}_{JF}(u)$ by considering only two types of malicious nodes, *BlackHole* and *JellyFish* nodes, for simplicity. Then we have the following proposition:

Proposition 1: Given a node u with d neighbors, i.e., $D(u) = d$, node u is isolated from the network if $\hat{n}_{BH}(u) \geq 1$ or $\hat{n}_s(u) + \hat{n}_{JF}(u) + \hat{n}_f(u) = d$.

Let Y_1 denote the event that a node u is isolated, $\hat{n}_g = \hat{n}_s(u) + \hat{n}_{JF}(u) + \hat{n}_f(u)$, by *Proposition* 1, we can obtain the probability that node u being isolated, given $D(u) = d$, as

$$Prob(Y_1|D(u) = d) = Prob(\hat{n}_{BH}(u) \geq 1|D(u) = d) \\ +Prob(\hat{n}_g(u) = d|D(u) = d). \qquad (1)$$

To be concisely, we omit the notation u in the following derivations. Let P_{BH} denote the probability of a node being a *BlackHole*, then the first item in (1) can be obtained as:

$$Prob(\hat{n}_{BH} \geq 1|D = d) = 1 - (1 - P_{BH})^d. \qquad (2)$$

The second item in (1) can be obtained by:

$$Prob(\hat{n}_g = d|D = d) = (1 - P_c - P_{BH})^d, \qquad (3)$$

where P_c is the probability of a node being cooperative, which can be obtained by the semi-Markov node behavior model proposed in [16]. By combining (2) and (3), we can rewrite (1) as:

$$Prob(Y_1|D = d) = 1 - (1 - P_{BH})^d + (1 - P_c - P_{BH})^d. \quad (4)$$

The effects of P_c and d on the probability of node isolation are shown in Fig. 2. In the figure, we can see that the probability of node isolation is inversely proportional to P_c given the fixed node degree, and may be proportional to d if P_c is fixed. Although increasing the node degree is a typical method to guarantee the network connectivity, we can see clearly, from the analysis above, that increasing the node degree may impact the network connectivity due to the potential DoS attacks launched by malicious nodes. Therefore, for MANETs, especially for military networks, mitigating node misbehaviors and thwarting DoS attacks are more challenging than topology control and network management.

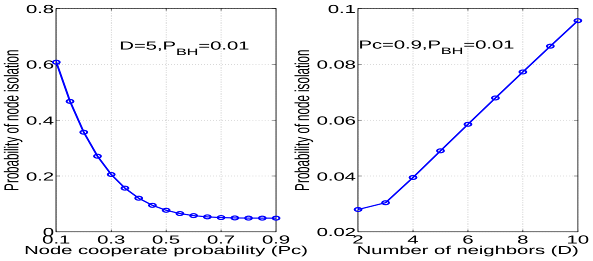


Fig. 2. Effects of P_c and d on node isolation probability.

IV. Dynamic DoS Attacks and Impacts

In the existing literature, DoS attacks such as *BlackHole* or *JellyFish* are studied without considering the node mobility or potential attack propagation, thus they are called *static* DoS attacks. However, malicious nodes may be able to move around the entire network, to adjust transmission power dynamically, or even to propagate DoS attacks by compromising their cooperative neighbors. Therefore, the DoS attacks may become *dynamic* in terms of the expansion of attack coverage and the propagation of attack impact. In this section, we introduce *dynamic DoS attacks* by beginning with several examples, then reveal its devastating impact on MANETs.

A. Examples of Dynamic DoS Attacks

1) Dynamic DoS Attack Using Node Mobility: The impact of DoS attacks may be spread by the mobility of malicious nodes. As shown in Fig. 3 (Left), a malicious node m attacks its three neighbors v_1, v_2 and v_3 first. After node m prevents the communications between its neighbors and other cooperative nodes, node m may move to another place, as shown in Fig. 3 (Right), continuing to launch DoS attacks against its new neighbors. If the malicious node u moves into an area with a higher node density, then more cooperative nodes may become the victims of DoS attacks.

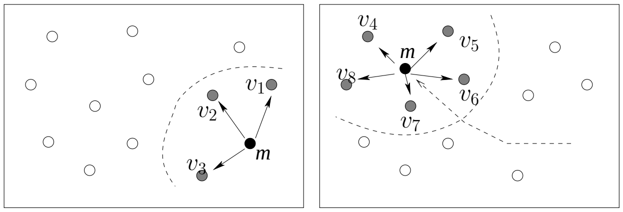


Fig. 3. DoS attack enhanced by malicious nodes movement.

2) Dynamic DoS Attack Using Power Management: When malicious nodes have the ability to adjust their transmission powers dynamically, then they can change their transmission ranges to enlarge the attack coverage. For example, in Fig. 4, a source node s needs to communicate with a destination node d. Then node s sends route discovery requests to its neighbors. When a malicious node m receives the forwarded request message, it can immediately increase its transmission power such that it can reach node s in one hop by increasing transmission range from R to R'. Next node m can unicast a route reply message to node s and claim itself only one-hop away from the destination d. This is a variant of BlackHole attack but more aggressive in that it affects the cooperative nodes beyond one-hop neighborhood.

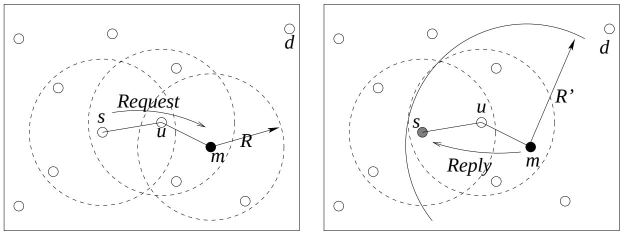


Fig. 4. DoS attack enhanced by dynamic power control.

3) Dynamic DoS Attack Using Worm-like Propagation: We believe that a malicious may be even able to compromise other cooperative nodes by probing vulnerability and sending some self-executable codes, such as *worms*. A malicious node can compromise its neighbors, then these compromised neighbors become interior attackers. Further, these compromised nodes may be used to compromise their neighbors continuously. By this way, DoS attacks can spread to a large area of the network or even the entire network, which is shown in Fig. 5.

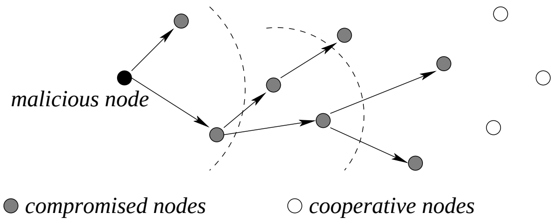


Fig. 5. DoS attack propagation by compromising immediate neighbors.

Instead of compromising the immediate neighbors, a malicious node may take the advantage of its cooperative neighbors to forward its malicious codes to the nodes of two-hop away, as shown in Fig. 6. By this selective compromisation, a malicious node can propagate DoS attacks even faster. In a more severe case, cooperative nodes are isolated with each other, while malicious nodes and newly compromised nodes can communicate via these isolated cooperative nodes. In other words, adversaries may deploy an overlay network on the original network efficiently by propagating DoS attacks dynamically and selectively.

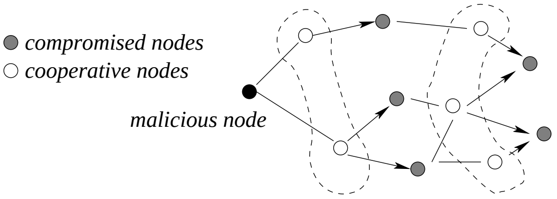


Fig. 6. DoS attack propagation by compromising non-adjacent nodes.

In order to evaluate the impact of dynamic DoS attacks, we introduce a simple attack propagation model right next.

B. Dynamic DoS Attack Propagation Model

To evaluate the propagation of dynamic DoS attacks, we want to know how likely and how fast a cooperative node can become malicious, and how long a malicious node will stay in a network to launch DoS attacks. Thus, we use a simplified version of the semi-Markov node behavior model proposed in [16] as the dynamic DoS attack model to facilitate our analysis in this paper. In our model, a node may operate at three states, i.e., *cooperative, malicious*, and *failed*, which comprise a state set $\mathcal{S} = \{C, M, F\}$. When a cooperative node is compromised, it transits from cooperative to malicious state. A cooperative node can become failed directly due to energy depletion. We assume that no mechanism is used to turn a malicious node into a cooperative one, so a malicious node will become failed at last once it runs out of energy. While, in our model, we assume that a failed node can become cooperative again after battery recharging. Hence, the behavior transition process described above can be defined by a semi-Markov process $\{Z(t)\}$ with state space $\mathcal{S}$. In the process $\{Z(t)\}$, the transition of states follows an *embedded Markov chain* and the transition time between two successive states may not be distributed exponentially. Let p_{ij} and T_{ij} be the transition probability and transition time from state i to j, respectively, for $i, j \in \mathcal{S}$, then the process $\{Z(t)\}$ can be described by a transition probability matrix $\mathbb{P} = (p_{ij})$ and a transition time distribution matrix $\mathbb{F}(t) = (F_{ij}(t))$. $\mathbb{P} = (p_{ij})$ and $\mathbb{F}(t) = (F_{ij}(t))$ are given by:

$$\mathbb{P} = \begin{pmatrix} 0 & p_{cm} & p_{cf} \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \ and$$

$$\mathbb{F}(t) = \begin{pmatrix} 1 & F_{cm}(t) & F_{cf}(t) \\ 1 & 1 & F_{mf}(t) \\ F_{fc}(t) & 1 & 1 \end{pmatrix}, \qquad (5)$$

where $F_{ij}(t)$ is the cumulative distribution function (CDF) of T_{ij} for $i, j \in \mathcal{S}$. The semi-Markov attack model is presented in Fig. 7.

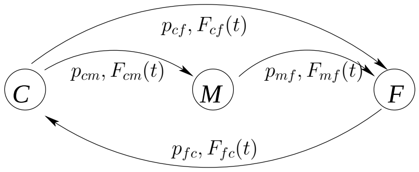


Fig. 7. Semi-Markov attack propagation model.

We are interested in the stochastic property of the propagation rate of dynamic DoS attacks, so we only address the transition time distributions in this paper. In particular, we focus on the distributions of the transit time from cooperative to malicious state, and the transit time from malicious to failed state, i.e., $F_{cm}(t)$ and $F_{mf}(t)$.

In fact, $F_{mf}(t)$ is the distribution of a malicious node's lifetime. We recall that in reliability engineering, the lifetime distribution of a system is widely modeled by Weibull distribution [17]. Thus we use Weibull distribution for $F_{mf}(t)$. To determine $F_{cm}(t)$, we consider the fact that port probing is a commonly used method by attackers to find the vulnerabilities of a system before the system can be compromised. As port probing progresses, less and less unscanned ports are left, then the potential vulnerabilities are more likely to be found. This process is similar as the process that a system's residual lifetime decreases gradually as less and less energy left and the system is failed once energy depletes. Thus, $F_{cm}(t)$ can be treated as a lifetime distribution and defined by Weibull distribution as well.

The Weibull function used in this paper is known as the two-parameter Weibull distribution, defined as

$$\mathcal{W}(\alpha, \beta) = 1 - \exp(-(t/\beta)^{\alpha}), \tag{6}$$

where α and β are usually called the *slope (or shape)* parameter and *scale* parameter, respectively. It is known that the mean of the Weibull distribution can be presented as $\mu = \beta\Gamma(1 + 1/\alpha)$, where $\Gamma(\cdot)$ is the gamma function. Thus, let μ_{cm} and μ_{mf} be the mean of T_{cm} and T_{mf}, respectively, then we have

$$\begin{aligned} F_{cm}(t) &= \mathcal{W}(\alpha, \mu_{cm}/\Gamma(1 + 1/\alpha)) \\ F_{mf}(t) &= \mathcal{W}(\alpha, \mu_{mf}/\Gamma(1 + 1/\alpha)) \end{aligned} \tag{7}$$

To validate our analysis, we use NS2 (v2.28) and MATLAB (v7.0) to perform the simulations. The simulation area is set to $500 \times 2000m^2$, on which 200 nodes with transmission range of $150m$ are distributed uniformly and randomly. IEEE 802.11 and AODV are used for medium access control and routing protocol, respectively. The total simulation time is set to $2000s$. The initial energy of each node is set to $10Joule$ which provides each node a lifetime around $20s$ (from the simulation results). When a node runs out of energy, its energy is reset to the initial energy level again after a recovery time delay averaged in $1s$. All nodes are set to be cooperative initially, while 2 of them are randomly selected to become malicious after the simulation begins. Once a node becomes malicious, one of its neighbors will become malicious after an average time of $2s$, which imitates the process of a malicious node compromising its cooperative neighbor and the compromised node continuing the compromisation.

From the simulation results, the average transition time from cooperative to malicious state and that from malicious to failed state are $7.13s$ and $8.76s$, i.e., $\mu_{cm} = 7.13s$ and $\mu_{mf} = 8.76s$, respectively. If let $\alpha = 2$, then we have $\beta_{cm} \approx 8$ and $\beta_{mf} \approx 10$ by using $\mu = \beta\Gamma(1 + 1/\alpha)$. We depict the distribution of T_{cm} from simulation results and the Weibull function $\mathcal{W}(2, 8)$ in Fig. 8, and depict the distribution of T_{mf} with the Weibull function $\mathcal{W}(2, 10)$ in Fig. 9. From Fig. 8 and 9, we can see that when these parameters ($\alpha = 2, \beta_{cm} = 8, \beta_{mf} = 10$) are chosen, the Weibull function in (7) match very well with the simulation results. The CDF plots show clearly how likely a node is compromised after a certain time. Further, the distribution can also be used to estimate the number of compromised nodes. For example, in Fig. 8, the probability that a node becomes malicious within $10s$ is almost 0.8, which also implies that 80% of nodes will become malicious within $10s$ if they are compromised. In the next section, we will discuss how fast DoS attacks may propagate in realistic ad hoc networks based on the model proposed in this section and some previous studies on worms.

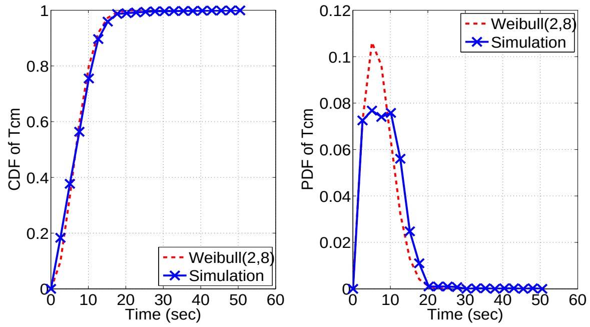


Fig. 8. Distribution of T_{cm}.

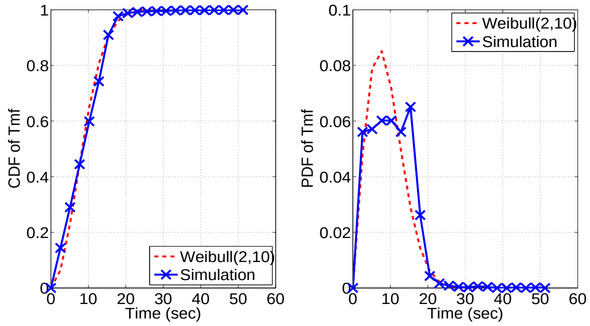


Fig. 9. Distribution of T_{mf}.

C. Discussions on Dynamic DoS Attack Propagation

As we mentioned in Section IV-A, besides launching DoS attacks, a malicious node may also compromise other nodes by using malcodes (like worms). For example, the Code-RedII worm [18] installs backdoors on the infected machines such that the infected machines could be used as "zombies" for future DoS attacks. In [19], it was shown that it takes only 2 to 5 minutes in average for an attacker to find out known vulnerability of a system. Consider the transmission rate in ad hoc networks is usually much lower than that in wired network, we conservatively assume that it takes 5 minutes in average for a malicious node to compromise its neigbhors, i.e., $\mu_{cm} = 5$ mins. If the malicious node has enough information to compute a spanning tree rooted at itself over the entire network, called *propagation tree* in the following context, then DoS attacks can be propagated from the root to all leaves. If the propagation tree is binary and balanced, then we know the lower bound of the tree depth is $\Theta(\log_2 N)$ if the system size is N. So we have an asymptotic propagation time as $\mu_{cm} \cdot \Theta(\log_2 N)$, for example, for a network with 1000 nodes, DoS attacks may propagate to the entire network within one hour ($5 \times \log_2 1000 < 60$). If the propagation tree is not balanced or has a long branch, then DoS attacks need longer time to spread in this case. As an extreme case, the tree is a *chain*, then the asymptotic propagation time is $\mu_{cm} \cdot \Theta(N)$, which may longer than the maximum lifetime of mobile nodes. If the propagation tree is not limited to binary, then the depth of the tree may be even shorter then $\Theta(\log_2 N)$, so that the propagation speed increases in this case. In general, if the network is d-regular or the average degree of the network is d, then we have the propagation time as $\mu_{cm} \cdot \Theta(\log_d N)$ asymptotically.

Since it is more likely for a malicious node to compute a balanced or multi-branch tree in a dense network than a sparse network, we know that the propagation of DoS attacks may be faster in a dense network than that in a sparse network even the dense network has a larger system size. Further, a malicious node in the central part of a network may impact the connectivity more severely than that in the border of the network because it can have more neighbors to be compromised. However, we notice that the propagation speed may slow down when more and more nodes are compromised. To explain this, we consider that as long as a cooperative node is compromised, it can launch DoS attacks as described in Section III. Therefore, these accumulated DoS attacks may impact network connectivity severely and isolate more and more nodes. By this way, the propagation of DoS attacks can "starve" its own propagation when the number of compromised nodes saturates to a certain level. Nevertheless, if malicious nodes compromise their cooperative neighbors selectively and wisely, like two-hop away only, then malicious nodes can exploit cooperative nodes to propagate malicious codes, such that DoS attacks can be spread without slowing down the speed.

We notice that mobility can mitigate the impact of static DoS attacks, for example, once a cooperative node moves out of the attack scope of a malicious node (with or without knowing the existence of the malicious node), then this node can continue communications with other nodes. Nevertheless, the mobility of a malicious node may be higher than other cooperative nodes such that a malicious node can enlarge its DoS attack scope by travesing a network.

V. Conclusion

In this paper, we first described the static DoS attacks on the link layer and network layer in MANETs. Then we analyzed the node isolation problem resulting from DoS attacks, and derived the probability of node isolation consequently. By considering the mobility of malicious nodes and potential propagation of DoS attacks, we introduced a new DoS attack, called *dynamic DoS attack*, by examples. These examples illustrated how dynamic DoS attacks can be launched by malicious nodes that take the advantage of power control or high mobility. The examples also shown how dynamic DoS attacks may propagate by compromising cooperative nodes. To evaluate the propagation speed of dynamic DoS attacks, we introduced an attack model based on a simple semi-Markov process and shown that the transition time from cooperative to malicious state can be defined by a Weibull distribution. After the analysis the dynamic DoS attack propagation, we found that: if the average time to compromise a node is denoted μ_{cm}, for a network of N nodes with an average degree of d, then the time of propagating a DoS attack to the entire network is $\mu_{cm} \cdot \Theta(\log_d N)$. The result implies that dynamic DoS attacks spread much faster in dense networks than in sparse networks given the fixed network size N, and for a constant node density the speed does not decrease substantially even if the network size N increases greatly. As a conclusion, the dynamic DoS attack is a more challenging issue for resilient MANETs design.

References

[1] V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks," in *Proc. of IEEE MILCOM '02*, 2002, pp. 1118 – 1123.

[2] P. Kyasanur and N. Vaidya, "Detection and Handling of MAC Layer Misbehavior in Wireless Networks," in *Proc. of IEEE Dependable Systems and Networks*, 2003, pp. 173 – 182.

[3] S. Radosavac, N. Benammar, and J. S. Baras, "Cross-Layer Attacks in Wireless Ad Hoc Networks," in *Information Sciences and Systems*. Princeton University, 2004, pp. 1266–1271.

[4] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks," in *Proc. of IEEE INFOCOM '03.*, March 2003.

[5] ——, "Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols," in *Proc. of ACM WiSe 2003.*, September 2003.

[6] I. Aad, J.-P. Hubaux, and E. W. Knightly, "Denial of Service Resilience in Ad Hoc Networks," in *Proc. of ACM MobiCom '04*, 2004, pp. 202–215.

[7] J. V. E. Molsa, "Increasing the DoS Attack Resiliency in Military Ad Hoc Networks," in *Proc. of IEEE MILCOM '05*, 2005, pp. 1 – 7.

[8] Q. Gu, P. Liu, S. Zhu, and C.-H. Chu, "Defending Against Packet Injection in Unreliable Ad Hoc Networks," in *Proc. of IEEE GLOBECOM '05*, 2005.

[9] S. Desilva and R. V. Boppana, "Mitigating Malicious Control Packet Floods in Ad Hoc Networks," in *Proc. of IEEE WCNC '05*, 2005, pp. 2112 – 2117.

[10] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad hoc Networks," in *Proc. of ACM MobiCom '00*, 2000, pp. 255–265.

[11] J. Kong, X. Hong, Y. Yi, J.-S. Park, J. Liu, and M. Gerla, "A Secure Ad-hoc Routing Approach using Localized Self-healing Communities," in *Proc. of ACM MobiHoc '05.*, 2005.

[12] Y. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A Secure OnDemand Routing Protocol for Ad Hoc Networks," in *Proc. of MobiCom '02*, Atlanta, USA, Sept. 2002.

[13] A. D. Wood and J. A. Stankovic, "Denial of Service in Sensor Networks," *IEEE Computer Magazine*, vol. 35, no. 10, pp. 54–62, Oct 2002.

[14] S. Xu and T. Saadawi, "Revealing the Problems with 802.11 Medium Access Control Protocol in Multi-hop Wireless Ad Hoc Networks," *Elsevier Journal of Computer Networks*, vol. 38, no. 4, pp. 531–548, 2002.

[15] G. Noubir and G. Lin, "On Link Layer Denial of Service in DATA Wireless LANs," *Wiley Journal on Wireless Communications and Mobile Computing*, August 2004.

[16] F. Xing and W. Wang, "Modeling and Analysis of Connectivity in Mobile Ad Hoc Networks with Misbehaving Nodes," in *Proc. of IEEE ICC '06.*, 2006.

[17] W. Q. Meeker and L. A. Escobar, *Statistical Methods for Reliability Data*. John Wiley and Sons Inc., 1998.

[18] D. Moore, C. Shannon, and J. Brown, "Code-Red: A Case Study on the Spread and Victims of an Internet Worm," in *IMW '02: Proc. of the 2nd ACM SIGCOMM Workshop on Internet Measurment*, 2002, pp. 273–284.

[19] F. Stevens, T. Courtney, S. Singh, A. Agbaria, J. F. Meyer, W. H. Sanders, and P. Pal, "Model-Based Validation of an Intrusion-Tolerant Information System," in *Proc. of 23rd IEEE International Symposium on Reliable Distributed Systems (SRDS'04)*, Oct. 2004, pp. 184–194.