

Vulnerabilities in Epidemic Forwarding

Alaeddine El Fawal, Jean-Yves Le Boudec, and Kave Salamatian

EPFL, I&C

CH-1015 Lausanne, Switzerland

{alaeddine.elfawal, jean-yves.leboudec, kave.salamatian}@epfl.ch

Abstract—We identify vulnerabilities in epidemic forwarding. We address broadcast applications over wireless ad-hoc networks. Epidemic forwarding employs several mechanisms such as inhibition and spread control and each of them can be achieved using different alternative methods. Thus, vulnerabilities existence is highly dependent on the used methods. We examine the links between them. We classify vulnerabilities into two categories: Malicious and rational. We examine the attack impact according to the number of attackers and the different network settings such as density, mobility and congestion. We show that malicious attacks are hard to achieve and their impacts are scenario dependent. In contrast, rational attackers always achieve significant gain. The evaluation is carried out using detailed realistic simulations over networks up to 1000 nodes. We consider static scenarios as well as vehicular networks.

I. INTRODUCTION

Epidemic forwarding has been proposed for use in wireless ad-hoc (i.e. infrastructure less) networks to disseminate information in the broadcast mode [4], [6], [14], [15]. The principle is that nodes repeat with some probability the information they hear from others, thus propagating fresh information around. It is designed to operate in quickly varying environments, where mobility and self-organization make the classical methods based on distribution trees are impractical.

In this paper, we identify vulnerabilities in epidemic forwarding over ad-hoc networks. We are interested in broadcast applications, where authentication is a very hard issue. Such applications can be chatting in a traffic jam, coupon advertisements [5] or pocket switched networks [13].

Epidemic forwarding employs several mechanisms. One of these mechanisms is inhibition that prevents a node from forwarding over-sent or over-received packets in order to minimize redundancy. Another mechanism is control of spread (number of nodes that receive a packet) control. Each mechanism can be achieved using different alternative methods. Thus, the existence of vulnerabilities is highly dependent on the mechanisms employed and on the methods adopted to achieve them. We examine the links between these methods and the vulnerabilities.

We classify the vulnerabilities into two categories: malicious and rational. With the former, an attacker harms other nodes and does not look for personal benefit. It aims at decreasing other nodes throughput and/or spread. In contrast, a rational attacker aims at increasing its personal profit of the network. It tries to increase its throughput or save battery and buffer.

We evaluate the vulnerabilities by simulations. We show that a malicious attacker does not have much effect in highly mobile networks but it might be very harmful in static networks. On the contrary, the rational attacker's benefit is independent of the mobility. It increases dramatically its throughput. Further, attacks that are otherwise very harmful lose their efficiency in

the presence of some epidemic forwarding mechanisms such as adaptive spread control and adaptive inhibition. Moreover, many elements such as the attacker position and the node density influence the malicious attacker's impact.

The simulations are carried out on networks with up to 1000 nodes using a JAVA implementation of the epidemic forwarding system in JIST-SWANS [1]. Beside static scenarios, we applied the epidemic forwarding system to the vehicular network using an extension of JIST-SWANS called STRAW [2], which provides a mobility model based on the operation of real vehicular traffic.

II. EPIDEMIC FORWARDING MECHANISMS

In this section, we explain the different mechanisms used in epidemic forwarding in order to understand their vulnerabilities. The natural way to do epidemic forwarding is Flooding. It consists of forwarding only once every received packet. It suffers from a huge amount of redundancy: to achieve a high delivery ratio a packet does not need to be forwarded by all nodes, but a small fraction of nodes is enough, in particular in dense network, and any other duplication is useless. Thus, inhibition mechanisms are needed. They prevent a node from forwarding over-sent or over-received packets in order to minimize redundancy. Furthermore, spread control mechanisms are essential as the broadcast capacity does not scale with the population. Also, a scheduler has been proposed to ensure some level of fairness and to do buffer management. And last, the rate of fresh packet injection by the application should be controlled according to the different conditions of the networks (e.g. dense, sparse and congested.)

A. Inhibition Mechanisms

We classify inhibition mechanisms into two sets: rigid and adaptive. With the former set, the mechanisms cannot adapt themselves to different network settings: when the settings change, their parameters need to change. The latter indicates the mechanisms that ensure a good performance in a wide range of settings without changing their parameters.

1) *Rigid Inhibition*: Within this set we find Gossip-based epidemic forwarding [8] where a node decides to forward a packet with a fixed probability p and drop it with $(1-p)$. The value of p is setting dependent but Gossip does not involve any mechanism to adapt it.

2) *Adaptive Inhibition*: within this set we distinguish between two methods.

a) *Counter Based Inhibition*: The first is essentially the one proposed in [9]. A packet stored in the epidemic buffer has a counter "Receive Count" incremented by 1 when a duplicate of this packet is received. Initially, i.e. when the packet is created by the application or received for the first time, the counter is

0. When the counter reaches a maximum value, the packet is discarded from the epidemic buffer. When a packet is transmitted, the value of Receive Count is lost. This method improves on classical TTL by discarding packets that are probably not worth transmitting.

b) Virtual Rate Based Inhibition: This method was proposed in [4]. With this method, a packet in the epidemic buffer is retransmitted with a probability that depends on its “virtual rate”; it is equal to $c_0 a^{Rb^S}$ where c_0 is a constant (inverse of a time), R [resp. S] is the number of times this packet or a duplicate was received [resp. sent] and a and b are unitless constants less than 1. Thus the virtual rate of a packet decreases exponentially with any receiving or sending event of the same packet. A scheduler decides which packet is selected next for transmission by the MAC layer; it serves packets per IP source fairly but with a rate not exceeding its virtual rate. The constant c_0 is equal to ηR_0 where R_0 is the nominal bit rate of the MAC layer and η is the fraction of time that the MAC layer spends serving epidemic packets (as opposed to packets of other, non epidemic applications).

B. Spread Control Mechanisms

We consider the following alternative methods for limiting the spread of packets.

1) Classical TTL: This is the method that comes by default with the Internet Protocol (IP). When a packet is created by a source and placed into the epidemic buffer, it receives a TTL value equal to some positive constant “max TTL”. When the packet is accepted for transmission by the MAC layer, the TTL field of the *transmitted* packet is equal to the value of the TTL field in the packet in the epidemic buffer, minus 1. The TTL field in the packet stored in the epidemic buffer is unchanged.

When a packet created by some other node is received for the first time at this node, the packet is delivered to the application, and the value of the TTL is screened. If it is equal to 0, it cannot be retransmitted and the packet is discarded. Else ($TTL \geq 1$), the packet is stored in the epidemic buffer, with TTL equal to the value present in the received packet. When and if the packet is later accepted for transmission by the MAC layer, the transmitted TTL field is equal to the stored TTL minus 1, and the stored TTL is unchanged, as above.

There is an issue when a duplicate of an existing packet is received, as is common with epidemic forwarding. In theory, there are 4 possible strategies: (1) keep the existing stored TTL (2) adopt the received TTL (3) keep the largest of the received and stored TTLs and (4) keep the smallest. However, strategies (2) to (4) are not practical as they are not resilient to simple bugs or attacks; with (2) and (3) a packet could keep a maximum TTL value for ever; with (4) a system could inject packets with $TTL=1$ and destroy existing packets. Therefore, only strategy (1) is implemented in practice and this is the one we consider in this paper.

2) Aging: This method was proposed in [4] in a different but essentially equivalent form. We give here a presentation that combines different options in one single framework. The method uses the TTL field like Classical TTL and its variants above, but the TTL of a packet may be decremented while it is stored in the epidemic buffer, depending on receive and send events. Formally,

every packet in the epidemic buffer has an “age” field, which is a fixed decimal positive number less than 256. When a packet, created by some other node, is received by this node for the first time, its age is set to the complement to 255 of the received TTL: $age = 255 - TTL$. When a packet is transmitted, its stored age is incremented by a fixed amount K_0 and then its TTL is set to $255 - age$. When a duplicate packet is received, the received TTL is ignored but the stored age is incremented: $age = age + K_1$. When *any* packet is received, the stored age of *all* packets in the epidemic buffer is incremented by K_2 : $age = age + K_2$. The node drops packets with age larger than 255.

C. Scheduler

Epidemic forwarding needs a scheduler for buffer management. To our knowledge, the only scheduler that is explicitly detailed in the literature is in [4]. It is used with the virtual-rate based inhibition (Sect. II-A.2.b). It decides which packet in the epidemic buffer is selected for transmission, i.e. to be passed to the MAC layer. Furthermore, in order to ensure some level of fairness, the scheduler serves packets per source Id, using a processor sharing approach. Moreover, every packet should be served at a rate not exceeding its virtual rate computed in Sect. II-A.2.b.

D. Control of Injection Rate

The only explicitly defined method to achieve control of injection rate is the one in [4]. It is used together with the aforementioned scheduler. The packets generated by the application at a given node are placed into the epidemic buffer, where they compete with the other packets for transmission (but with a high probability of being transmitted as their virtual rate is larger, having $R = S = 0$). The application rate is controlled by a windowing system : The number of outstanding packets the application is allowed to have in the epidemic buffer at this node is limited to -at most- 2 [4]; a packet is deleted from the epidemic buffer when a duplicate is received, which serves as implicit acknowledgment (Ack).

III. ATTACKS

In this section, we describe the vulnerabilities that are specific to epidemic forwarding. We distinguish between two types of attackers: malicious and rational. The former does not look for a personal benefit but aims to harm other nodes. In contrast, the latter seeks to increase its personal profit from the network. The most of the attacks are described by drawing (Figs. 1 and 2) using a generic example where the attacker is M and the victim in malicious case is A.

A. Malicious Attacks

A malicious attacker aims at decreasing the spread of the victim by exploiting vulnerabilities in epidemic forwarding mechanisms. In the following we identify five attacks and map them to their corresponding epidemic forwarding mechanisms.

1) Artificial High Density (AHD): In this attack, we exploit the adaptability of the spread control to the congestion and node density. The attacker places himself close to the victim. It acts like any node: it has its self packets (packets that are generated at this node) to send and relays others packets. But it does not forward victim packets. By generating much traffic in the very close surrounding of the victim, the attacker incites the spread-control mechanism at the victim’s good neighbors to react negatively and prevent the victim packets from going farther.

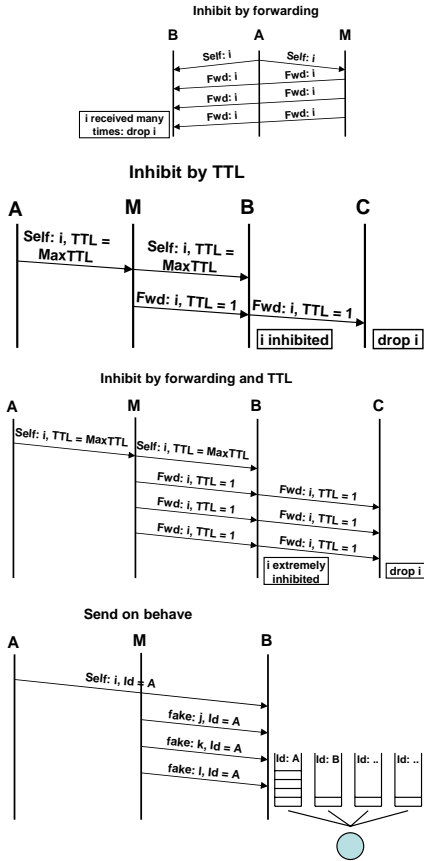


Fig. 1. Malicious attacks: M is the malicious node and A is the victim. We refer by Self to packets generated at the node transmitting them, by Fwd to packets forwarded by the node but generated by others and by Fake to packets that are generated by the malicious node, M, but carrying the victim identity (Id = A). To understand the figures, we will explain only the "Inhibit by forwarding and TTL" attack as other attacks have similar explanation. In "Inhibit by forwarding and TTL", A sends a Self packet i with $TTL = \text{maxTTL}$, that is received by M and B. M forwards the packet i (Fwd: i) 3 times with $TTL = 1$, the packet (Fwd: i) is received by B and C. Thus, the inhibition mechanism at B will inhibit packet i since it is received 4 times and C will drop the packet since it is $TTL = 1$ and it can not be forwarded.

2) *Inhibit by Forwarding (IbF) Attack*: With IbF (Fig. 1), the attacker exploits the adaptive inhibition. It forwards the victim packets immediately a number of times, called Attack-Persistency, to inhibit its neighborhood from forwarding the same packets (see Sect. II-A.2). With the counter based inhibition (see Sect. II-A.2.a), the Attack-Persistency is equal to the maximum value the counter can reach. With the virtual rate based inhibition, this Attack-Persistency should be large enough to make the corresponding virtual rate close to zero (in practice two times are enough).

3) *Inhibit by TTL (IbTTL) Attack*: This attack exploits the spread control using TTL. As the attacker receives a victim packet, it forwards it immediately with a $TTL = 1$. In Fig. 1, B and M receives the a packet from A with $TTL = \text{MaxTTL}$. M forwards it with $TTL = 1$ instead of $(\text{MaxTTL}-1)$. Hence, the attacker decreases the chance the packet has to travel beyond C as B is inhibited and C drops the packet. Even if B succeeds in forwarding the packet after M, this will change nothing with C. Note that in Fig. 1, B applies the first strategy (see Sect. II-B.1) upon receiving a duplicate of the packet and thus it keeps the old TTL. If it applies the fourth strategy, IbTTL will be more

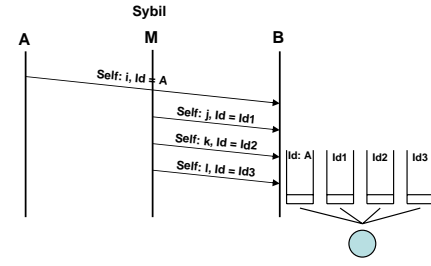


Fig. 2. Rational Sybil attack: M is the rational node.

harmful since both nodes, B and C, will drop the packet upon receiving it with $TTL = 1$.

4) *Inhibit by Forwarding and TTL (IbFTTL) Attack*: This is a combination of IbF and IbTTL (Fig. 1). In this case M forwards the victim packets Attack-Persistency times with $TTL = 1$ to insure that the victim packets at B are well inhibited and thus the packet loses any chance to travel beyond C.

5) *Send on Behalf of the Victim (SoB) Attack*: SoB, the attacker exploits the scheduler and the aging mechanism. In Fig. 1, M sends fake packets with A's Id. As the scheduler serves packets per source Id to ensure fairness, A's packets are delayed in the epidemic buffer and they will be dropped either by the aging mechanism (they become too old) or by buffer overflow.

B. Rational Attacks

We identify two rational attacks specific to epidemic forwarding.

1) *Do Not Cooperate (DNC) Attack*: When a new packet is injected by the application at a given node, it is placed in the epidemic buffer, where it competes with packets received from other nodes. This competition prevents the application from injecting at the full rate allowed by the packet injection control mechanism because of the additional delay in the epidemic buffer. Thus, an attacker decides to not cooperate and to keep only its self packets (packets that are generated at this node) in the epidemic buffer. Note that, if the attacker tries go beyond the allowed rate, its packets will be accumulated in other nodes, which are not able to serve at the same rate. Thus it risks killing its packets because of the same reason as explained in Sect. III-A.5.

2) *Sybil Attack*: We refer to the Sybil attacker as the node that forges multiple identities [3]. This a well-known attack in networking but the way it is exploited in this paper is new and very specific to epidemic forwarding. As the scheduler serves packets per source Id, the attacker sends its self packets with different Ids and thus it increases their service time. In Fig. 2, we present the scheduler as a process sharing approach where queues are per source Id. In this case, M's packets receive more service time than A's packet at B.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the impact of the aforementioned attacks by simulation. We apply them to static scenarios, as well as to highly mobile networks. We consider vehicular mobility on the highway. Our metrics are based on the spread and rate: a malicious attacker aims at reducing the victim spread and a rational one tries to increase its rate while maintaining large spread.

In our simulation we consider the epidemic forwarding system proposed in [4], called SLEF. To our knowledge, SLEF is the only complete system proposed for a wide range of settings. Furthermore, SLEF implements all epidemic forwarding mechanisms already discussed in Sect. II: The virtual rate based inhibition, spread control by TTL and aging, injection rate control and the scheduler discussed in Sect. II-C. The parameter values of the virtual rate based inhibition are $a = b = 0.15$, c_0 corresponds to 802.11b basic rate (1Mbps). As for the aging, we use $K_0 = K_1 = 25$ and $K_2 = 0.5$.

A. Settings

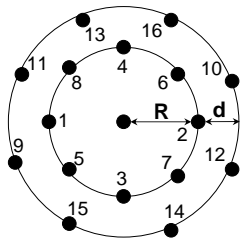


Fig. 3. Malicious attackers places: The victim is in the middle, attackers start filling the places around the victim according to their index in an increasing order: if one attacker, it fills position 1. If 2 attackers, they fill position 1 and 2, and so on. R can be either 25m or 100m. $d = 25$ m.

With the static scenarios, we simulate from 200 up to 600 nodes uniformly distributed over a square of 500×500 m², but, in most cases, we show the results only for 400 nodes as the others are similar. The transmission range is around 50 m (PDA transmission range).

In the case of a malicious attack, the victim is in the middle of the square and attackers take place around it as it is indicated in Fig. 3. We want to evaluate the impact of the distance between attackers and the victim. Therefore, the radius R in Fig. 3 can have one of two values: 25m and 100m. With the former, the attackers of the corresponding circle are within the transmission range of the victim and they are outside it with the latter.

In the case of a rational attack, there exists only one attacker, which is in the middle.

The network can be either congested, where all nodes are sources sending at full rate (capacity allowed by the channel) or non-congested, where the victim is the only source in the network and it is sending at full rate. Beside the victim, only attackers can act as sources in the non-congested scenario, based on the attack they want to achieve.

In the following we will use the following notations: “close” (“far” respectively) to indicate that R is equal to 25m (100m respectively) and “one source” (“all sources” respectively) to indicate that the network is non-congested (congested respectively).

As for the mobile scenario, we simulate vehicles in an urban road with two lanes. The speed limit is 80 km/h. The car density is 12.5 cars/km in each direction. We simulate 1000 nodes. The transmission range is 300m, which is typical for vehicular network.

Our simulations are carried out through JIST-SWANS [1], an open source simulator for ad hoc networks. The MAC layer is a very accurate implementation of 802.11b in DCF mode with

the basic rate of 1 Mbps as we transmit in broadcast (pseudo-broadcast [4]). As for the radio, we use the capture effect to approach the real WIFI cards, which all implement it [7]. We consider fading channels with free space path-loss. As for the mobile network, we use an extension of JIST-SWANS called STRAW [2], which simulates the vehicular traffic and provides a mobility model based on the operation of real vehicular traffic.

B. Static Scenarios

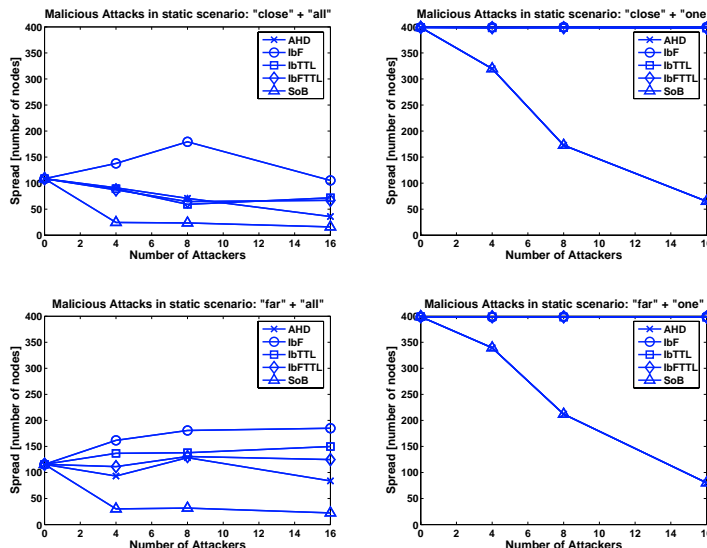


Fig. 4. Malicious attacks in static scenario. “close” = attackers are within transmission range with the victim. “far” is the opposite of “close” (see Sect. IV-A). “all” = all nodes are sources. “one” = the victim is the only source. The x-axis shows the number of attackers. The y-axis shows the spread in number of nodes that receive a packet. The network contains 400 nodes.

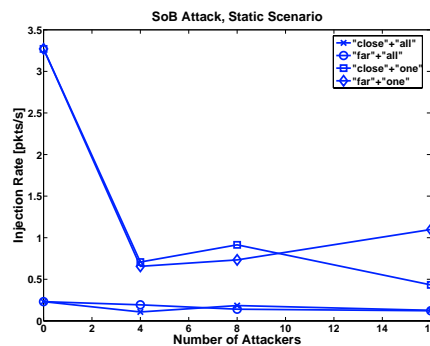


Fig. 5. Rate of a victim facing SoB attack. The x-axis shows the number of attackers. The y-axis shows the rate of the victim. The network contains 400 nodes.

1) Malicious Attacks:

a) *AHD*: The results are shown in Fig. 4. Let us start with the “all-sources” scenario where the attackers are sources and act as any other node except that they do not forward victim packets. In the “close” case (see Sect. IV-A), the impact of the attack is considerable and it increases with the number of attackers. In contrast, in the “far” case, the attackers increase the spread instead of reducing it. This can be explained by two reasons: (1) the attackers are far from the victim and thus they do not increase the density as much as in the “close” scenario; (2) the inhibition mechanism is adapted. Indeed, the attackers are numerous and they cooperate in forwarding all packets except the victim ones, hence they inhibit their neighbors from forwarding packets except

those of the victim. Thus, the increase in the victim spread is due to the fact that victim packets are less inhibited than others. If the inhibition were rigid, we expect that AHD would have more impact on the victim. In the “one-source” scenario, attackers are still injecting new packets in the network as before.

b) *IbF*: Now, the attackers do not generate fresh packets, their role is merely to forward victim packets as it is explained in Sect. III-A.2. The results are shown in Fig. 4. It is clear that *IbF* does not achieve its goal. This is due to (1) our implementation of the attack and (2) the implicit Ack (acknowledgment) used by SLEF to do injection rate control. In our implementation of *IbF*, when an attacker receives a new victim packet, it forwards it immediately (if the MAC layer allows) Attack-Persistence times. If the same attacker receives another victim packet before it finishes forwarding the previous packet, it cancels the previous and it starts anew with the newest. Thus, let us consider a scenario that happens frequently. The victim sends a new packet. The attacker forwards it immediately. The victim receives a duplicate of its self packet and considers it as an implicit Ack. Hence, it injects a new self packet that will be received by the attacker before finishing the forwarding process and by other attackers even before beginning the forwarding process. Thus, all attackers cancel the previous packet, which explains why it is not inhibited. Another strategy to implement the attack is to not begin forwarding a new victim packet before finishing from the previous one. This strategy is worse for the attacker because the new packet will have the chance to escape from the attacker barrier before the attackers even begin forwarding it.

c) *IbTTL*: Our implementation of *IbTTL* is similar to the one of *IbF* with the difference that it modifies the TTL before forwarding, as it is explained in Sect. III-A.3. This attack is more harmful than *IbF*. The attacker needs to forward the packet only once with $TTL = 1$. Thus, nodes that receive the packet from the attacker for the first time are not able to forward it because of its TTL. This makes the difference with *IbF*, which needs to forward several times to inhibit the packet in its neighborhood.

d) *IbFTTL*: This attack has approximately the same impact as *IbTTL*, which is to be expected as *IbF* has little effect on the victim.

e) *SoB*: Now, the attackers send only fake packets at full rate. Fig. 4 shows a significant decrease in spread and rate. The spread reduction is due to the fact that victim packets are killed in the epidemic buffers before being forwarded because of the delay caused by the fake packets (for more explanation see Sect. III-A.5). Moreover, the decrease in rate is due to the delay of the implicit Ack that controls the injection rate as it is explained in Sect. II-D.

From what we have seen in this section we can conclude that the attackers are not able to harm the victim in the presence of mobility because of two reasons. The first is that the impact of the attackers is very position dependent. The second is that, even with the most harmful attack, the attackers could reduce the spread of the victim but its packets still reach few tens of nodes. If these nodes are mobile, they will carry the victim packets beyond the barrier imposed by the attacker. This conclusion is well verified later in the vehicular network scenario.

C. Rational Attacks

f) *DNC*: We evaluate the impact of *DNC* only in the “all-sources” scenario, where increasing the rate is a challenge. In

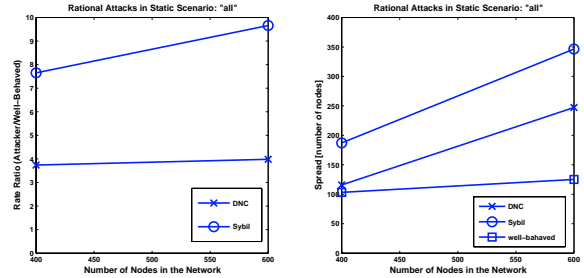


Fig. 6. Rational attacks in the static scenario. There is only one attacker in the network. The spread and the rate of the attacker are compared to those of a well-behaved node that is very close to it and thus facing the same network conditions. The y-axis on the left shows the rate ratio between a the attacker and the well-behaved node. On the right, the y-axis shows the spread. The x-axis in both shows the number of nodes in the network.

the “one-source” scenario, the attacker is the only source in the network and he has the entire network capacity, thus it is meaningless to evaluate its impact in this case. In Fig. 6, the performance of a *DNC* attacker is compared with a well-behaved node that is very close to it and thus both experience the same network conditions. We show the spread of both nodes and the rate ratio (*DNC* over well behaved). The *DNC* rate is four times larger than a well-behaved node. But, surprisingly, the *DNC* spread is much larger when the network is very dense (600 nodes). The reason is as follows: The attacker does not forward others packets. Thus, when it receives other packets than self packets, it drops them without updating the age of its self packets in the epidemic buffer. Hence, the age of the attacker self packets does not increase during their stay in its epidemic buffer by K_2 (see Sect. II-B.2), which allows them to travel farther. Note that, from the large gain in rate (four times larger), we can conclude that the amount of age lost for the attacker self packets is important, which is translated by a considerably larger spread.

g) *Sybil*: We evaluate the impact of *Sybil* in only “all-sources” scenario for the same reason as with *DNC*. The attacker uses five different identities. In addition, it does not forward others packets. So, our implementation is in fact a combination of both attacks, *Sybil* and *DNC*, explained in Sect. III. This implementation gives the attacker a much larger gain than using *DNC* alone (up to 10 times larger than a well-behaved node and 2.5 larger than the *DNC* attacker), which explains the impact of *Sybil* alone.

D. Vehicular Network Scenario

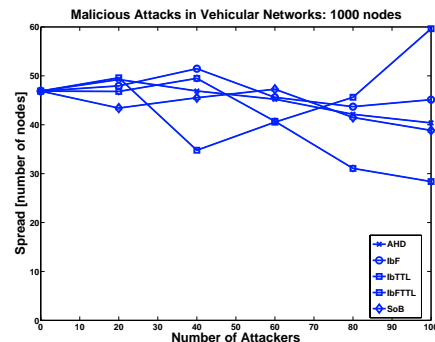


Fig. 7. Malicious attacks in the vehicular network. The spread of the victim is plotted according to the number of attackers in the network. Beside the attackers, the network contains 1000 well-behaved nodes.

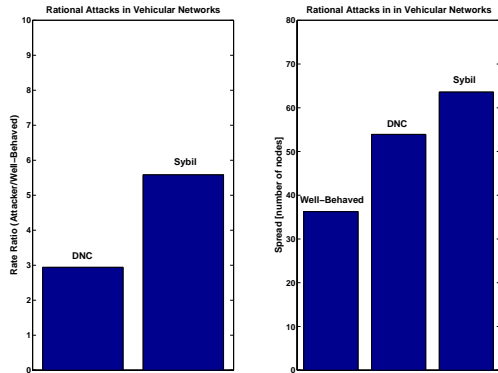


Fig. 8. Rational attacks in the vehicular network. The spread and the rate of the attacker are compared to the average of other well-behaved nodes. On the left, the rate ratio between the attacker and the average of well-behaved nodes is shown for both attacks. On the right, we show the spread. The network contains only one attacker and 1000 other well-behaved vehicles.

In this scenario, nodes are highly mobile and the position of the victim is not known. Thus, the attackers are chosen randomly. Beside the attackers, the network contains 1000 well-behaved nodes. All nodes are crossing the same urban road.

1) *Malicious Attacks*: Fig. 7 shows the impact of malicious attacks. The spread of the victim is drawn according to number of attackers. The Attack-Persistence of IbF and IbFTTL is 2. Other values give the same results. As we can notice, the impact of the attackers is negligible even in the presence of 100 attackers. In the most harmful case, the IbFTTL attacker reduces the victim spread from 50 to 30 nodes which is not significant. This can be explained by the presence of the spread control mechanism; the attacker can affect the victim only if their spreads interfere, i.e. there exist common nodes that receive the attacker and the victim packets. And the amount of harm is proportional to the amount of interference. Since the spread is limited by the spread control mechanism, this interference is not considerable and does not happen frequently.

2) *Rational Attacks*: Contrary to malicious attacks, rational attacks are still powerful even in highly mobile network. The results are shown in Fig. 8. Sybil still ensures higher gain than DNC.

V. STATE OF THE ART

To our knowledge, this is the first work that identifies vulnerabilities that are specific to epidemic forwarding, i.e. that use epidemic forwarding mechanisms such as inhibition, spread control injection rate control and scheduler.

Some of vulnerabilities that we identify could be recovered by cryptographic and authentication methods if they are available. But all already existing work in the literature on securing wireless network does not apply here since we address broadcast application over wireless ad-hoc networks. In particular, an extensive work assumes the existence of a third trusted part that is the infrastructure [11], [12], which does not exist in what we are doing. In [10], the authors proposed a method based on global synchronization to secure routing. Global synchronization assumes the presence of infrastructure unless mobiles are occupied with GPS (Global Positioning System) device that we don't consider in our work. Further, with the broadcast nature of the applications that we address, nodes do not need to know each

other to communicate and thus they can not trust each other. Till now and at our best knowledge, there does not exist any work proposing an authentication method for this scenario.

VI. CONCLUSIONS

We identified vulnerabilities that are specific to epidemic forwarding. We addressed broadcast applications over wireless ad-hoc networks. We classified vulnerabilities into two categories: malicious and rational. We evaluated their impact according to the number of attackers and the different network settings. We found that malicious attacks impacts depend on the position of the attacker relative to the victim, the network density, the traffic load and mobility. In static scenarios, some attacks could reduce dramatically the victim spread and rate whereas other attacks could not harm the victim due to the adaptive inhibition and the injection rate control. In highly mobile vehicular network, the impact of malicious attacks are minimized due to the spread control.

We studied the rational case in presence of only one attacker in the network. The attacker could achieve considerable profit in all scenarios.

Our work can be extended in different directions. We plan to examine the impact of the presence of several rational attackers on the network. Another extension is to find solutions to recover from these vulnerabilities.

REFERENCES

- [1] Java in simulation time / scalable wireless ad hoc network simulator , jist/swans, <http://jist.ece.cornell.edu/>.
- [2] Street random waypoint / vehicular mobility model for network simulations , straw, <http://www.aqualab.cs.northwestern.edu/projects/straw/>.
- [3] J. R. Douceur. The sybil attack. In *The 1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, March 2002.
- [4] A. El Fawal, J.-Y. Le Boudec, and K. Salamati. Self-Limiting Epidemic Forwarding. Technical Report LCA-REPORT-2006-126, EPFL, 2006.
- [5] A. Garyfalos and K. Almeroth. Coupons: Wide scale information distribution for wireless ad hoc networks. In *IEEE Global Telecommunications Conference (Globecom) Global Internet and Next Generation Networks Symposium Dallas, Texas, USA*, pages 1655–1659, December 2004.
- [6] Z. J. Haas, J. Y. Halpern, and L. Li. Gossip-based ad hoc routing. *IEEE/ACM Transactions on Networking*, 14(3):479–491, 2006.
- [7] A. Kochut, A. Vasani, A. U. Shankar, and A. Agrawala. Sniffing out the correct physical layer capture model in 802.11b, berlin, germany. In *IEEE International Conference on Network Protocols (ICNP 04)*, pages 252–261, October 2004.
- [8] S.-D. Modiano, E. and G. Zussman. Maximizing throughput in wireless networks via gossiping. In *ACM SIGMETRICS / IFIP Performance'06*, June 2006.
- [9] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. In *Mobicom, Seattle, Washington, United States, August 15 - 19, 1999*, pages 151–162.
- [10] A. Perrig, R. Canetti, D. Song, and J. D. Tygar. Efficient and secure source authentication for multicast symposium, NDSS'01. In *Network and Distributed System Security*, February 2001.
- [11] M. Raya, I. Aad, J.-P. Hubaux, and A. El Fawal. DOMINO: Detecting MAC layer greedy behavior in IEEE 802.11 hotspots. *IEEE Transactions on Mobile Computing*, 2006.
- [12] M. Raya, P. Papadimitratos, and J.-P. Hubaux. Securing Vehicular Communications. *IEEE Wireless Communications Magazine, Special Issue on Inter-Vehicular Communications*, 13(5):8–15, 2006.
- [13] J. Scott, P. Hui, J. Crowcroft, and C. Diot. Hagggle: A networking architecture designed around mobile users. In *IFIP WONS 2006, January 18-20, Les Menuires, France*.
- [14] I. Stojmenovic, M. Seddigh, and J. Zunic. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, Jan. 2002.
- [15] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks, 2000.