Attack Injection for Performance and Dependability Assessment of Ad hoc Networks

Jesús Friginal, Juan-Carlos Ruiz, David de Andrés, Pedro Gil

Attack Injection for Performance and Dependability Assessment of Ad hoc Networks [*]

Jesús Friginal, Juan-Carlos Ruiz, David de Andrés and Pedro Gil
Fault-Tolerant Systems Group (GSTF), Instituto de Aplicaciones de las TIC Avanzadas (ITACA)
Universidad Politécnica de Valencia, Campus de Vera s/n, 46022, Spain
Phone: +34 96 3877007 Ext {75774, 85703, 75752, 79707}, Fax: +34 96 3877579
{jefrilo, jcruizg, ddandres, pgil}@disca.upv.es

Abstract

Ad hoc networks are wireless, self-configuring and self-maintaining networks that allow dispensing of fixed infrastructures by relying on nodes cooperation for providing packet routing. Their confident use claims for the existence of methodologies for assessing their performance and dependability attributes in the presence of malicious faults. Most of the work deployed in this domain has been based on simulation, thus obviating aspects influencing the behavior of real ad hoc networks, like interferences. This paper reports a preliminary research carried out with the purpose of providing an experimental approach for deriving performance and dependability measures from real ad hoc networks. The goal is to show the feasibility of the technique, while providing a taste of the type of measures and conclusions that can be derived from the supported experimentation. This research constitutes a first step towards a more ambitious objective, the provision of a dependability and performance benchmarking approach for ad hoc networks.

1 Introduction

Nowadays ad hoc networks present a great potential in a variety of application domains ranging from military scenarios [10] to ambient intelligence environments [4]. Although ad hoc networking seems to hold the key of ubiquitous computing, its industrial exploitation claims for new approaches to assess performance and dependability.

Routing protocols are cornerstones of ad hoc networks. They are responsible for enabling communications in absence of fixed infrastructures. Nodes must cooperate and rely on each other to provide routing services. On the one hand, they act as end-hosts within their radio range. On the other hand, they behave as routers for other network nodes far apart. A link is a one-hop connection between two nodes. A set of links enabling the communication between a source and a destination defines a (multi-hop) *route*. Applications running inside network nodes communicate through *data flows* traversing network routes.

Recent research in ad hoc networks has focused on medium access control and routing problems, resulting in the proposal of many different routing protocols [1]. Attending to the method followed to establish network routes, routing protocols can be classified as reactive or proactive. Reactive routing protocols, like AODV, find a route on demand by flooding the network with Route Request packets. Conversely, proactive routing protocols, like OLSR, maintain fresh lists of destinations and their routes by periodically distributing routing tables.

A wide range of different measures has been defined so far in the literature [3]. As stated in [2] most of such assessment research is based on simulation studies, thus reporting results based on propagation and network model assumptions that are quite different from those existing in real life ad hoc environments. For the sake of representativeness, it is essential to complement existing simulation-based measures with field data retrieved from real ad hoc networks.

Ad hoc networks encompass security vulnerabilities that are vastly different from their traditional wired counterparts [5]. Their lack of infrastructure, dynamic topology, and openness of wireless links is responsible for this. Attacks against ad hoc networks manipulate sensitive information exchanged among nodes to establish communication routes. Such manipulation may result in network partitioning, might affect network data traffic or may introduce a certain overload. The evaluation of the impact of attacks on real ad hoc networks remains an open challenge.

This paper is structured as follows. Section 2 proposes a set of measures and an attack injection-based approach to cope with their computation. Section 3 reports on results analysis. Finally Section 4 presents conclusions.

[*] This work is sponsored by the Spanish project MEC TIN2006-08234.

2 Experimental approach

Experimentation will rely on a real ad hoc network that will act as experimental platform for supporting the proposed experimental approach (see Figure 1).

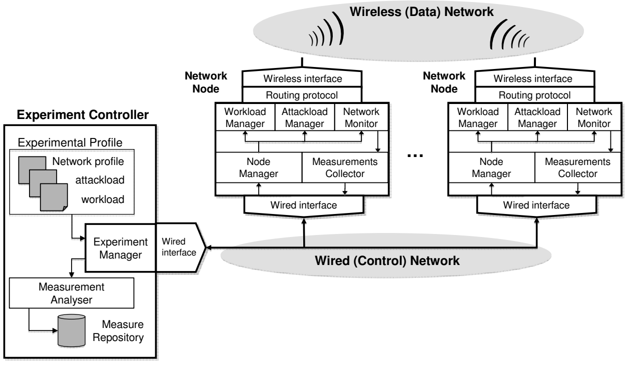

Figure 1. Experimental platform architecture.

Network nodes could act either as common nodes, which send and forward traffic to other nodes, or as malicious nodes (attackers), which are responsible for perpetrating attacks. For practical considerations, this approach considers the existence of two networks. The first one constitutes the experimental target, i.e. the ad hoc (wireless) network that nodes use to exchange information. The second one defines the control and coordination (wired) network. The use of a wired network avoids provoking interferences during experimentation with the wireless network. It also improves controllability over experimentation since prevents losing control of nodes out of the experiment controller radio range.

Although real wireless communications are considered, the mobility of nodes is emulated via mobility patterns. The experiment controller instruments each node to filter those packets sent by nodes out of its radio range, thus emulating nodes' visibility. As stated in [8], radio interferences caused by the absence of mobility of nodes during experimentation are negligible for networks with a limited number (few tenths) of nodes. Considering real mobile devices may arise other problems, like (i) the necessity of wider space for experimentation, (ii) the controllability of considered nodes, and (iii) how to make devices actually move in practice.

Regarding the benefits of the experimental platform, it is important to mention its openness and flexible architecture. The routing protocol is a component that can be changed to assess the attributes of different types of ad hoc networks. In the same way, the wireless interface can also be replaced to consider different types of physical wireless technologies.

Under these considerations, experiments are performed following a four-fold approach: (1) the experiment execution profile is defined and distributed to all network nodes; (2) the experiment is run, i.e. network nodes are activated and they behave according to their mobility pattern, workload and attackload; (3) once the experimental time elapses, the experiment ends and resulting measurements (activity logs) are retrieved from each node; (4) finally, measurements are transformed into measures and stored in the platform repository for a later analysis.

Following paragraphs detail the proposed measures, the execution profiles required for experiments configuration, the experimental procedure, and finally, the processing that derives measures from resulting measurements.

2.1 Measures

The approach estimates the impact of attacks in ad hoc networks using both performance- and dependability-related measures. Table 1 presents the set of measures that has been considered for the purpose of the present study.

Table 1. Proposed measures.

Performance measures	**Description**
Goodput	Data flow throughput, i.e. number of applicative bits per time unit traversing the network from a given source node to a certain destination. This measure does not consider the overhead induced in the network by ad hoc routing protocols and retransmitted data packets. The higher (in average) the better.
Packet loss	Rate of packets in data flows not correctly delivered by the network. The lower (in average) the better.
Route insertion time	Time required by a given node to integrate the ad hoc network. This is a measure characterizing the proneness of a routing protocol to integrate new nodes in the network. The lower (in average) the better.
Energy consumption	Average energy consumed by a network node during the experiment. The lower (in average) the better.
Dependability measures	**Description**
Data flow availability	Average rate for an applicative packet to be delivered from a given source node to a certain destination. It only makes sense when attacks affect a particular data flow, otherwise it will be equal to route availability.
Route availability	Average probability for any packet traversing a route to be delivered.
Route vulnerability	Average rate for a route to become the potential target of an attack. It is estimated through the percentage of time a route is in the coverage range of a malicious node. In our context, data flow vulnerability equals route vulnerability. It must be noted that we consider a route under attack as vulnerable.

2.2 Execution profile

The execution profile defines the parameters characterizing the ad hoc network and their nodes (network profile), the applicative traffic among the nodes (workload), and the attacks the network may experience (attackload).

Network profile: Network profiles must determine the scenario where experiments take place. This means defining the spatial distribution (topology) of network nodes and existing obstacles in the deployment area. Nodes' location changes along the time according to their speed and mobility pattern. Typical mobility patterns are Random Waypoint, Gauss-Markov, Manhattan Grid, and Reference Point Group Mobility models [9]. The wireless technology and the obstacles signals may find, determine the node's radio range. Configuration parameters of routing protocols, like neighbor hold time and refresh interval for OLSR, condition their behavior, mainly their detection and reaction capabilities in case of changes in communication routes.

Workload: The selected workload, i.e. the applicative traffic exchanged among ad hoc network nodes, should be representative of that of ad hoc networks deployed on real environments. This decision increases the representativeness of experimental results. That is the reason why this paper encourages the use of traffic generated by real applications, such as voice over IP (VoIP), FTP or email clients, instead of using synthetic traffic.

Attackload: The attackload emulates the behaviour of attackers in real ad hoc networks. Attacks can be perpetrated by external or internal attackers and can be categorized into passive or active. Active attacks may be further classified according to their execution mode into simple or complex. Simple approaches send topology control messages to target the network's topology. Malicious nodes can (i) forge and disseminate new messages, or (ii) collect old valid control messages describing an already non-existent topology and re-send them, causing other nodes to update their routing tables with fake routes. The effect of these approaches can range from degradation to denial of service. Complex approaches use the routing level to gain control over the applicative level. In this case, topology control messages are sent to change the course of a given data flow and ensure that it traverses now the malicious node. According to this, attackers may drop packets or modify their order or content. As an example, selfish nodes typically implement packet dropping-based attacks, like black or grey hole attacks. Malicious nodes perpetrating black hole attacks drop packets, thus breaking communication routes. Those implementing grey hole attacks drop only a particular type of packets, thus only affecting a concrete data flow. Interested readers can find in [6] [7] further information on how to inject such types of attacks.

2.3 Experimental procedure

An attack injection campaign comprises a certain number of experiments with exactly the very same execution profile. The experimental procedure followed in each of these experiments is depicted in Figure 2.

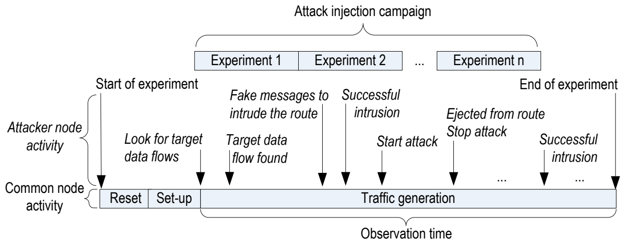

Figure 2. Experimental procedure.

Once the execution profile gathered, the experiment controller transfers the relevant information to each network node via the wired control network. During the set-up phase the initial network topology is established and network nodes are configured to play a specific role (as common or attacker nodes) during experimentation.

Common nodes just generate applicative traffic and forward all received packets.

The behaviour of attacker nodes is far more complex and follows the next process. Attack injection points cannot be precomputed due to nodes' mobility. So, attacker nodes must sniff the medium searching for packets of targeted applicative data flows being transferred between two common nodes. The link between these nodes is then dynamically selected as an injection point. After that, attacker nodes generate fake routing messages to intrude the targeted route. Once in the route, attacker nodes gain control over the route and all data flows traversing it. Hence, they can deploy any malicious action on controlled data flows. Eventually, after some time, common nodes in the targeted route may notice that something odd is happening and may try to conform an alternative route. In this case, attacker nodes may be ejected from intruded routes, thus losing their ability to perform attacks. It must be noted that such situation may be prevented if attackers try to keep their position in the route by sending fake routing information. Attacker nodes may also leave targeted routes as a result of nodes' mobility, i.e. whenever attackers fall out of the radio range of involved nodes. In any case, attackers may resume their malicious operations by sniffing the medium to locate a suitable data flow.

During the whole experimentation time, network nodes are in charge of monitoring their own activity. Common nodes' measurements are aimed at obtaining information about the generated traffic (required for computing goodput, packet loss and delay) and the availability of established routes. Attacker nodes monitor the occurrence of events related to their operations, such as the time when a target applicative data flow was detected (it becomes vulnerable) or a route was intruded (effective attack). At the end of the experimentation, all these data are transferred to the controller which will analyse them to determine the impact of injected attacks on the behaviour of the ad hoc network.

Table 2. Common node log.

```
1233226811.067381 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 110 ... udp/rtp 56 c73 * 7 803269919 848597774
1233226811.091705 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 87 ... udp/rtp 33 c3 35111 160 3825775171
1233226811.100480 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 87 ... udp/rtp 33 c3 35112 320 3825775171
1233226811.116171 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 87 ... udp/rtp 33 c3 35113 480 3825775171
...
1233226811.172287 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 1067 ... udp/rtp 1013 c31 63117 0 803269919 0x0000
1233226811.175453 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 1068 ... udp/rtp 1014 c31 63118 0 803269919 0x0010
1233226811.176811 00:21:00:02:46:66 > 00:1a:73:a1:62:e9, ethertype IPv4 (0x0800), length 1010 ... udp/rtp 956 c31 63119 0 803269919 0x0010
...
```

2.4 From measurements to measures

Three different logs are generated by the nodes while monitoring the behaviour of the ad hoc network during experimentation. These logs hold information related to the activity deployed by common nodes (*common node log*), attacker nodes (*attacker node log*) and ping requests (*ping log*). The whole set of data stored in these logs must be processed, correlated, and analyzed to extract those values needed to estimate the measures defined in Table 1. As shown in Figure 1 all this treatment is performed by the Measurement Analyzer of the Experiment Controller. The rest of this section details how expected measures can be deduced from all these experimental measurements.

An excerpt of the *common node log*, is shown in Table 2. Each line in this file represents information related to an applicative packet sent or received by a given node. The first element of each line is a timestamp, stating when this packet was sent or received. Maintaining the internal clocks of network nodes synchronized is essential for the correlation of the different logged timestamps. NTP (Network Time Protocol) services, for instance, can be very useful for this task. The following items in the log identify the MAC address of the source and destination nodes and the contents of the packet header. It is to note that each node involved in the transference of applicative traffic (workload), either as sender or receiver (or both), will generate this kind of log.

The common node log is processed in order to compute the *goodput* of a certain applicative data flow. Since goodput represents the applicative data received by a node, only the log of the destination node should be analyzed. Each log line is scanned looking for packets of a given data flow, sent by the source node to this destination. Once located, the length of each packet is used to compute the total amount of information successfully transferred. The goodput for that data flow will be computed as the relation between this amount of information (in bits) and the duration of the experiment (in seconds). The timestamp of received packets can be used to compute the goodput of a given data flow during particular periods of time. Furthermore, the evolution of the goodput along the time can also be carried out by computing the goodput in intervals of one second.

The *packet loss*, for a given data flow, can be perceived as the relation between the number of packets not reaching its destination and the total number of packets sent. Hence, in order to obtain these numbers, the logs in both the source and destination nodes should be analyzed. Those logs are scanned looking for packets of a given data flow, sent by the source node to the destination one. Then, each packet sent by the source node with an identifier matching that of a packet received by the destination node is marked as successfully transmitted. In this way, unmatched packets sent by the source node are considered as lost packets.

A similar process can be followed to compute the average *delay* of data flows. This delay is calculated as the difference between the timestamps of matched packets.

Since working with real devices, the *energy consumption* of each node can be estimated by comparing the initial and final battery power level for each experiment.

Although the *common node log* has shown its usefulness to compute performance-related measures, the other two logs (*the ping and attacker node logs*) are required to estimate dependability-related measures.

The *route availability* represents the average probability of a packet to be delivered from source to destination nodes. In order to estimate this probability it is necessary to deploy some mechanism in the ad hoc network to determine whether the communication between source and destination is possible at any time. Since the network workload does not ensure that applicative packets are continuously exchanged between nodes, ICMP ECHO REQUEST (ping) messages are continuously sent from source to destination. This activity is reported by *ping logs* (see Table 4). Each log's line represents a small slot of time the experimentation has been divided into. The first item in each line is a timestamp identifying that time slot. Several ping packets are sent from source to destination in each time slot. Thus, the second element of each line is a boolean value stating whether at least one packet was received by the destination node and, hence, the communication between the nodes was available. Availability will be then computed by the relation between the time in which communication was possible and the whole experimentation time. It must be noted that in absence of attacks, data flow availability equals route availability. This

Table 3. Attacker node log.

```
1233226824.839366 E1 DETECTION data flow between 192.168.2.56 and 192.168.2.55
1233226836.093937 E3 OFF INTRUSION data flow between 192.168.2.56 and 192.168.2.55
1233226836.116471 E2 GENERATING malicious routing protocol messages between 192.168.2.56 and 192.168.2.55
...
1233226858.660319 E3 ON INTRUSION data flow between 192.168.2.56 and 192.168.2.55
1233226858.684511 E2 GENERATING malicious routing protocol messages between 192.168.2.56 and 192.168.2.55
1233226859.797914 E4 ON DATA FLOW DISRUPTION between ports 5000 and 5099
...
```

may not be true in presence of malicious nodes implementing data flow disruption attacks, since the route may remain available for all data flows except the targeted one.

Table 4. Ping log.

```
1233226773.899245 1
1233226775.523744 1
1233226777.495388 0
1233226779.936998 0
1233226784.011103 1
...
```

Data flow availability is computed attending to the information provided by the attacker. As shown in Table 3, the *attacker node log* lists all "malicious" events it deploys on the network. The first element of each log's line is a timestamp stating when an event occurred. This is followed by an event identifier and description. Event E1 refers to the detection of a target applicative data flow. E2 relates to the generation of fake routing messages to either intrude or keep in a route. E3 indicates the state of the intrusion, which can be on (the route is intruded) or off. The same applies to E4 that reports on the state of the attack, which can be on (the attack is in progress) or off. It must be noted that these events are those grouped in Figure 2 under the denomination of attacker node activity. In the particular case of the attack log reported in Figure 3, the reader can see that the attack only disrupts packets (i.e. data flows) sent by node 192.168.2.56 to 192.168.2.55 in a port range between 5000 and 5099. More precisely, the disruption consists in packet dropping, so the log traces the activity deployed by a grey hole attack. The availability of a data flow is thus computed as the relation between (i) the time where the route is available and no attack is in progress (no E4 ON event occurs) and (ii) the total experimental time.

In order to estimate the *vulnerability of a given data flow or route* (remember that they are the same in our context), information regarding the network topology is necessary. Basically the location of nodes and their movement must be studied in order to determine the intervals of time where common nodes are in the radio range of attacker nodes, i.e. when they are vulnerable.

3 Preliminary results

This section reports on first results retrieved from experimentation on a real mobile ad hoc network.

3.1 Experimental set-up

The ad hoc network infrastructure provided by our experimental platform follows the general architecture defined in Figure 1. The platform comprises 7 Linux laptops acting as network nodes and a Linux PC assuming the role of the experiment controller. In each experiment, 6 laptops are common nodes and the remaining one behaves as an attacker. The wired network is an ethernet-based network and the wireless network relies on IEEE 802.11b as physical communication layer.

To cope with the mobility of real nodes, the experimental platform relies on the Castadiva test bed [8], which emulates nodes' mobility and their relative location. Castadiva was configured to randomly deploy nodes over an square area with a side of 700 m and their radio range was set to 250 m. The movement of the nodes around this area was randomly generated by following a Random Way Point (RWP) mobility pattern with a speed ranging from 1 to 3 m/s.

10 different attack injection campaigns, of 50 experiments each, were defined. Different initial deployments and mobility scenarios were generated for each campaign to compare the results obtained from different configurations. However, all the experiments within the same campaign shared the same configuration in order to increase the confidence on the computed results.

The protocol selected for experimentation was the Optimized Link State Routing (OLSR) protocol, a proactive ad hoc routing protocol. Its specific implementation was the olsrd v.0.4.10 [11].

The selected workload consisted in three data flows with different source and destination nodes. Two of them were Voice over IP (VoIP) data flows, whereas the third one was devoted to socket communications.

The attacker node was configured to perform grey hole attacks on VoIP data flows as attackload.

Experiments presented a set-up time of 50 seconds that was devoted to the configuration of the ad hoc network nodes. After this time, the workload was run for another 200 seconds. This experiment duration was selected in order to keep the generated log files under reasonable sizes. Each experiment was executed twice, once in absence and once in presence of attacks defined by the attackload. Determining the impact of grey hole attacks on the behaviour of the considered ad hoc network can be achieved by comparing the results provided by these two executions.

3.2 Analysis of measures

Analyzing the goodput of the data flows targeted by grey hole attacks is a good starting point to determine the impact of attacks on the network behaviour. Figure 3 depicts the computed goodput for the two considered VoIP data flows.

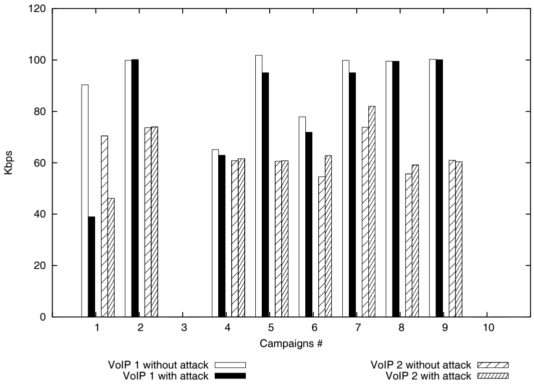

Figure 3. Impact of grey hole attacks on the data flow goodput of considered campaigns.

As can be seen, the goodput of these data flows in absence and in presence of attacks only differs significatively in campaign #1. In the rest of campaigns, the goodput does not present significative differences. Probably, due to particular characteristics of the randomly selected initial topology and movement pattern of these campaigns, the attacker node was not able to perpetrate a successful attack. In this sense, campaigns #3 and #10 are a very special case, since their initial topology and evolution do not allow the ad hoc network to establish communication between the nodes involved in the VoIP traffic. The study of the rest of estimators will later confirm this initial hypothesis.

It is interesting to take a closer look to the goodput results obtained for a particular campaign. Figure 4 depicts the average goodput for each of the experiments comprising the campaign #1. It can be easily seen that, on average, the goodput decreases around 60 kbps in the presence of attacks. The intersection of the lines representing the goodput in absence and in presence of attacks denotes that the attack was not likely to be effective in these experiments (experiments #10, #12, #16, #36, and #42, for instance). Taking a closer look at the results obtained in a single experiment could help us to deduce the effect of the attacker on the network behaviour. For instance, the dotted line in Figure 5 shows that the goodput of the considered data flow becomes 0 in presence of an attack just before 150 seconds of experimentation. It could be possible to guess that this sharp drop of the goodput is caused by the malicious operations deployed by the attacker node. Nevertheless, the accidental or malicious origin of this effect requires further information.

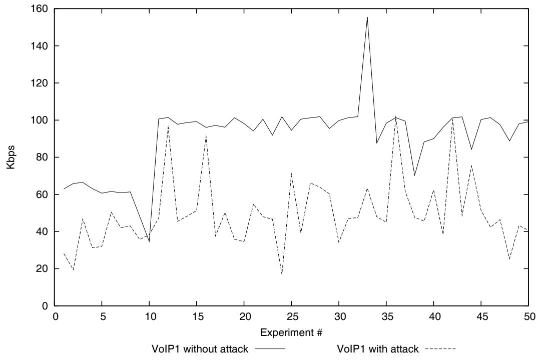

Figure 4. Impact of grey hole attacks on the goodput of experiments in a given campaign.

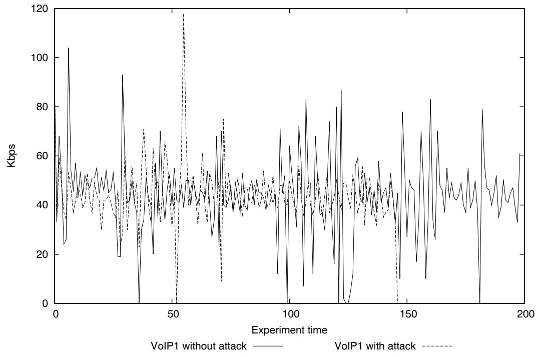

Figure 5. Impact of grey hole attacks on the goodput of an experiment along the time.

Computing the packet loss rate contributes to discerning whether malicious actions have taken place in the network. Since all experiments in a campaign present the same network profile, they all should present a similar packet loss rate. However, as Figure 6 shows, campaign #1 is the only one presenting a significative increase (20% and 45% respectively) in the packet loss rate exhibited by both targeted VoIP data flows. Determining whether an attacker has provoked such increase requires a more detailed analysis.

As stated, the data flow availability defines the probability for a particular data flow to reach its destination suc-

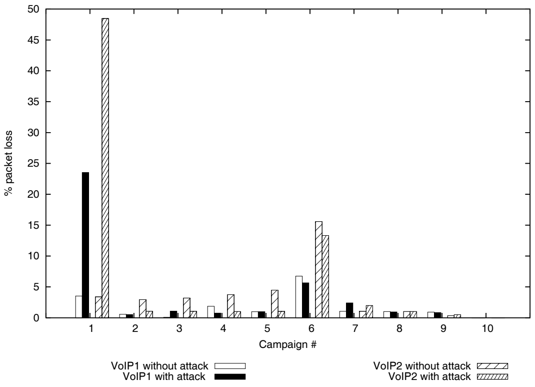

Figure 6. Impact of grey hole attacks on the packet loss rate of considered campaigns.

cessfully. This value may decrease due to the node's mobility, which will prevent communication from establishing, or due to the operation of a malicious node, which drops that particular traffic. However, in the case of attacker-free experiments, availability should only be related to the nodes' mobility. Hence, the comparison of the availability obtained in absence and in presence of attacks, will provide an estimation of the impact of attacks on the behavior of the ad hoc network. The obtained results are depicted in Figure 7.

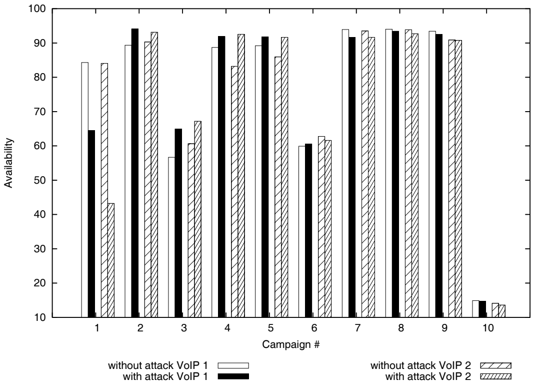

Figure 7. Impact of grey hole attacks on data flow availability of considered campaigns.

In all the considered campaigns, but campaign #1, the difference between the two computed values is not significative. These results seem to confirm the previously stated suspiction: the goodput decrease and the packet loss rate increase of experiments in campaign #1, in presence of attacks, could be directly attributed to the attacker.

A precise representation of the effect of a grey hole attack over a particular VoIP data flow in one experiment is provided in Figure 8. It correlates data flow goodput, and route and data flow availability with information regarding the vulnerability into the same graphic. As can be seen in Figure 8, there exists a goodput fall around second 50. The zone defined before that time is labelled as Z1. As can be seen Z1 become Z2 when the VoIP data flow and route availability becomes 0. Zone Z2 denotes an interval of time where a node has left the route. This goodput fall may be attributed to accidental causes. After that, the network is reconfigured and both the route and the data flow become available once more (Z3). Then, the data flow becomes vulnerable when an attacker is in its radio range. This situation is depicted by zone Z4, where despite the presence of the attacker both the considered route and data flow remain available. However, in Z5 the attacker has already intruded the route and has started a grey hole attack affecting the data flow. As can be seen, the route continues available for all data flow but the targeted one (whose availability is 0). Due to mobility reason, after some seconds, the attacker comes out of the radio range of its target, thus entering zone Z6, where as in Z2, neither the route nor the data flow are available. Once the communication re-established, the zone Z7 is entered. In Z7 the network resumes its normal operation, i.e. both the considered route and data flow become available. This example underlines the importance of measure correlation and its powerfulness for analyzing what is really happening in the network when an attack is injected.

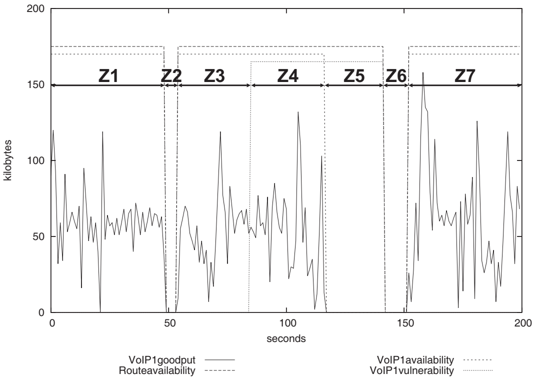

Figure 8. Data flow goodput versus route and data flow availability and vulnerability.

At this level it is important to underline that only 56 out of 500 attack injection experiments were really useful for the analysis of the ad hoc network behavior in the presence of attacks. This result derives from the great influence that the selected topology and network nodes movement have on the probability of success of considered attacks. In other words, in many experiments carried out, the mobility pattern of nodes has not given the attacker a chance to intrude any network route. This is a matter of the amount of time a route remains continuously in the attacker radio range (zone

Z4 of Figure 8). Computing this *route insertion time* for all performed experiments results in an average of 30.92 seconds. As can be seen in Figure 9 only attackers in campaign #1 (representing the 10% of the total experiments) are in their victims' range for around 90 % of the experimental time. A deeper analysis of this campaign has showed that, 45 out of 50 experiments resulted in a successful grey hole attack; 8 of them presented an interesting attack pattern where two successive attacks were carried out by the same attacker. This was due to the fact that, due to mobility reasons, between the first and second attack, the victim's route has fallen out of the attacker radio range. In these experiments, the attacker was able to maintain its attack active for 20.82 seconds on average. Regarding campaigns #6 and #7, it must be underlined that, although the attacker is in the radio range of its victims for about 20% of the experimentation time, campaigns barely present successful attacks since routes are not continuously exposed to the attacker for more than the required route insertion time.

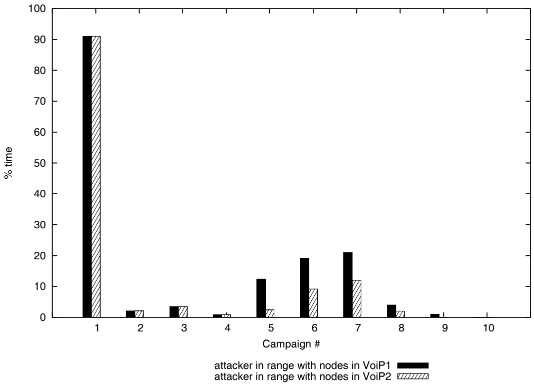

Figure 9. Percentage of time the attacker is in radio range of the targeted nodes.

According to our results, the considered attack does not drain significatively the battery of nodes in the attack network route since, both in absence and in presence of attacks, laptops consumed between 20 and 22 Watts per second.

4 Conclusions

Although in previous papers we have already explored how to inject black hole [6] and grey hole attacks [7] in OLSR-based ad hoc networks, this is our first publication reporting field data extracted from a systematic experimentation carried out on a real ad hoc network.

Results seems quite promising. We have identified and quantified those network attributes required to increase the effectiveness of attack injection campaigns. The goal is not to provide guidelines for hacking such networks but rather identify new vulnerabilities and, at the same time, obtain as many useful information as possible with the same experimental time. We have also illustrated how performance and dependability measures can be correlated in order to quantitatively characterize in detail the behavior of the network in presence of attacks.

The approach proposed in this paper opens a wide set of new experimental possibilities that can help us to increase our knowledge on how different types of real ad hoc networks behave in practice. This is an essential previous step towards the definition of useful approaches for benchmarking the dependability and performance of ad hoc solutions. In addition, obtaining concrete information about attacks is also interesting for designing new, and improving existing, intrusion detection and tolerance mechanisms. We claim these points will gain in importance as ad hoc networks become more common in our everyday lives thanks to the development of ambient intelligence-based solutions.

References

[1] I. Chlamtac et al. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks*, Vol. 1(1):13–64, 2003.

[2] J. Choi and Y. Ko. A performance evaluation for ad hoc routing protocols in realistic military scenarios. In *Int. Conf. on Cellular and Intelligent Communications*, Korea, 2004.

[3] S. Das and J. Yan. Simulation based performance evaluation of mobile, ad hoc network routing protocols. In *ACM Mobile Networks and Applications Journal*, pages 179–189, 2000.

[4] G. Cabri et al. The LAICA Project: An Ad-hoc Middleware to Support Ambient Intelligence. *Multiagent and Grid Systems Journal*, Vol. 2(3):235–247, 2008.

[5] Hao Yang et al. Security in mobile ad hoc networks: challenges and solutions. *IEEE Wireless Communications*, Vol. 11(1):38–47, 2004.

[6] J.-C. Ruiz et al. Black Hole Attack Injection in Ad hoc Networks. In *IEEE Dependable Systems and Networks (Fast Abstract), Proc. Supplemental Volume*, USA, 2008.

[7] J.-C. Ruiz et al. Towards Measuring the Security of Ad-hoc Routing Protocols. In *IEEE SRDS, AMBER Workshop, Proc. Supplemental Volume*, Italy, 2008.

[8] J. Hortelano et al. Castadiva: A test-bed architecture for mobile ad hoc networks. In *IEEE Symp. on Personal, Indoor and Mobile Radio Communications*, pages 1–5, 2007.

[9] D. Johnsson and D. Maltz. Dynamic Source Routing in Ad hoc Wireless Networks. *Mobile Computing book by Kluwer Academic Publishers*, pages 153–181, 1996.

[10] M. Natu and A. Sethi. Adaptive fault localization in mobile ad hoc battlefield networks. *IEEE Military Communications Conference (MILCOM)*, Vol. 2:814–820, 2005.

[11] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). *RFC 3626*, 2003.