A Study of On-Off Attack Models for Wireless Ad Hoc Networks

L. Felipe Perrone, *Member, IEEE,* and Samuel C. Nelson

Abstract— **The security of mobile wireless ad hoc networks is a multifaceted topic which in recent years has been the focus of much interest in the research community. While many security issues in these networks can be addressed by protocol design, wireless nodes have inherent physical vulnerabilities that can be exploited by attackers to cause disruptions in network traffic. The nature of these exposures is such that there is little that can be done to eliminate them leaving the network open to *denial of service* and *reduction of quality* attacks. It is essential that the impact of such attacks is well-understood before wireless ad hoc networks are used in mission-critical applications. This paper is a step in this risk analysis as our experiments quantify the effects of attacks that exploit physical vulnerabilities. Our contributions in this paper are twofold. First, we introduce a general model that can be used to characterize a physical attack as an *on-off* process and then we apply this model to two specific attack scenarios. Second, we present the results of a simulation study with these two attack scenarios in the context of a mesh of network nodes using the IEEE 802.11 standard and the AODV routing protocol.**

Index Terms— **Modeling, simulation, wireless ad hoc networking, security.**

I. Introduction

THE development of the technology for wireless ad hoc networks has picked up a strong momentum in the last few years. A wide range of applications for this technology has been identified including remote sensing, surveillance, and communications. Wireless ad hoc networks promise to create self-configuring backbones that enable communications in the absence of infrastructure and without need for human intervention.

One of the most important benefits associated with this technology is that nodes are free from any wires that would constrain their ability to move. The network nodes operate on battery power and communicate with each other via radio. The nodes' freedom of movement is constrained only by their need to stay connected to the collective – one may not stray too far from the rest so that it can establish a radio link to at least one peer. The flexibility that comes from wirelessness allows computation and communication to reach localities where cable deployment is problematic or even inviable.

The protocols that drive wireless ad hoc networks allow them not only to be self-configurable, but also to be resilient to failure. As nodes move around, some radio links break while new ones are established. The Layer 3 or routing protocols used in wireless ad hoc networks are constructed to allow nodes to adapt themselves to changing circumstances. When a link breaks and a route fails, the routing protocol attempts to discover another path from source node to destination node. Some times this recovery process is successful, some times it is not.

The technologies for network Layers 1 and 2 have developed fast in comparison with Layer 3. Today, IEEE 802.11 wireless networking, for instance, is well-established and radio devices for this standard are inexpensive and widely available. Layer 3 protocols for wireless ad hoc networking, on the other hand, are not yet nearly as mature and are still the subject of much research and development.

One of the most relevant issues in the construction of a Layer 3 protocol for wireless ad hoc networking is security. The technology's potential to be a key resource in many mission critical applications can only be fulfilled when the network can be made resistant, if not impervious, to attacks that hinder its operation. The problem of securing routing, already a hard one to solve in wired networks, is magnified by an order of magnitude in wireless networks, which rely on an inherently insecure, broadcast medium. We illustrate it briefly as follows.

Abstracting away the details of the technology, one can look at the network as an directed graph $G = (V, E)$, where each node $v_i \in V$ is connected to some neighbor $v_j \in V$ by an edge $(v_i, v_j) \in E$ and possibly by a reverse edge $(v_j, v_i) \in E$. The traffic of packets from node v_i to another node $v_k \in V$, which is not a neighbor of v_i, must be forwarded by neighbors of v_i along a sequence of nodes until reaching the destination. This sequence of nodes $\langle v_i, \ldots, v_k \rangle$ determines a path in graph G. These forwarding paths are determined by the Layer 3 protocol, which inspecting the list of neighbors of v_i, the list of neighbors of these neighbors and so on, can explore transitivity relations to identify the shortest route connecting the source v_i to some destination v_k.

An attack on a routing protocol focuses at disrupting the establishment of routes. Often, the attack scenario determines that a malicious node v_e advertises to its neighbors false information. A common scenario is that of the "black-hole attack" in which v_e advertises to its neighbors that it is directly connected to any other node in the network. The neighbors, believing that they can reach all destinations at most in two hops, change their routing tables to forward traffic to v_e expecting that it will push it along to the destination, when in fact it doesn't. This kind of attack is only possible when v_e can convince other nodes that it is indeed an authorized participant of the network. Proposals for securing Layer 3 protocols against this kind of attack often rely on enforcing *data confidentiality* in the wireless communication channels with the use of data encryption, and also by requiring that participant nodes *authenticate* themselves with one another

[1], [2], [3]. Other kinds of attacks on Layer 3 protocols are possible and their countermeasures may require the establishment of additional security properties at the level of the packets exchanged by communicating nodes such as *freshness*, *ordering*, *integrity*, and *non-repudiation*.

There remains, however, the fact that an additional security property not included in the discussion above, namely *availability*, is key in determining the successful operation of the application which utilizes the network. The attacker in a wireless ad hoc network can make attempts on its availability in a variety of ways; the ultimate goal is to prevent the network from playing its role as a communications medium. Attacks that compromise availability are often categorized as *denial-of-service* attacks [4], which attempt to prevent the network from performing its function as it would normally be expected.

Our goal, in this paper, is to establish a flexible, simple, simulation model for attacks on availability, and to show its application in two different scenarios. The model we present is based on the premise that attacks have only two possible states: either the attack is *on*, meaning that some action is being performed on a specific node, or the attack is *off*, meaning that the node is left to operate as normal. We formalize this model by defining it as a composition of arbitrary stochastic processes.

It is important to point out that, while it may have wider applicability, our model has been designed to represent *simple* denial-of-service (DoS) and reduction-of-quality (RoQ) attacks. By *simple* we mean situations in which attackers do not adapt their actions to react to changing values of network performance metrics or to exploit specific protocols executed in the network.

We illustrate the application of this simulation model by using it to explore two different kinds of attacks on a mesh network built with the *Ad hoc On-demand Distance Vector* (AODV) [5] routing protocol running on the IEEE 802.11b wireless standard. The attacks we consider are such that do not require technical sophistication on their perpetrator and demand the physical capture of one or more network nodes. Once a node is captured, it is subjected to actions that have the side effect of causing the node's network software to create additional control traffic.

The model we present consists of a combination of stochastic processes and is suited to characterize patterns where the attacker alternates periods of activity with period of inactivity. Our simulation experiments with a small mesh network (36 nodes) explore various points in the large parameter space of the overall model. The results we obtained indicate that a cyclical attack pattern with these simple attacks can leverage the adaptation transients in wireless protocols to substantially affect the performance of the network.

The remainder of the paper is organized as follows. Section II establishes the context for this work and compares what we have done to previous research efforts by other authors. Next, in Section III, we introduce a general simulation model for *on-off* attacks in abstract terms. We showcase the application of the model in Section IV, where we present an attack in which a node is physically captured and made to shutdown and later to reboot, and in Section V, where we present an attack in which the geographical coverage of a node's radio signal is forced to alternate between two values. Section VI presents the experimental analysis of a large set of simulations that explores points of the parameter spaces of the two attacks introduced previously and, finally, Section VII concludes the paper and points out directions for future work.

II. Related Work

There have been other proponents of models similar to what we discuss in this paper. The most notable of these is the *reduction-of-quality* (RoQ) attack model due to Guirguis et al. [6]. Their work describes an attack which acts as an additional source of network traffic on a TCP connection. The attack works as a square wave, that is, an *on-off* process, in which for a duration of τ seconds an additional M packets are pushed through an existing TCP connection. This work demonstrates that an attacker does not need to introduce an overwhelming amount of additional traffic into the network to produce significant disruption. The success of the periodic attack is due to the fact that the attack introduces transient conditions that are addressed by the network's adaptation mechanisms. When, in the process of adjusting to the new transient, the source of the attack enters a state of inactivity, the network must again adapt to new conditions.

The idea that motivated our work was developed independently, based on the same premises, but was specifically proposed in the context of wireless ad hoc networks by Chip Elliott and Stephen Boswell [7]. Wireless ad hoc routing algorithms, either pro-actively or reactively, create control traffic for the network in the effort to adapt to changing network conditions. When an existing link fails or when a new link is discovered, the network nodes to which this link is incident, may be led to consider the possibility that a new, shorter route has become available. Upon detecting a change in the status of one of the links along a path, say from operational to broken, the network layer protocol in the node initiates the traffic of control messages as dictated by its routing algorithm. This traffic, which should be present during a transient period, aims to discover routes and to correct forwarding tables in the nodes in the neighborhood of the changed link. However, if the status of the same link were to alternate periodically, bursts of routing control messages would be injected on the network. Depending on the frequency and the duration of these bursts, the network would observe degradation in performance metrics such as packet delivery ratio and end-to-end delay.

While a number of security problems in wireless ad hoc networking can be countered by algorithmic constructs at the protocol level, there are vulnerabilities that are harder to address. For instance, say that a group of individuals sets out to capture one network node each. They agree to coordinate their actions according to some loosely synchronized clock and, at the appointed time, each one disconnects the battery from the captured node. As each node is powered down, their neighbors eventually discover that they cannot route traffic through them (no acknowledgment for data transmissions is the indication that the node is out). At some later point in time, each attacker reinstalls the battery in the captured node, which powers up and goes about its own neighbor and route discovery all over again. The node outages disrupt data traffic. The routing algorithm's reaction to outages injects control traffic in the network, what can degrade its performance. This specific kind of attack may be prevented by tamper-proofing the network node, but as long as it is possible for one to act against the quality or the availability of the network's radio links, the network is still vulnerable.

There has been little investigation on the effects of attacks on wireless ad hoc networks beyond the level of attacks at the protocol level. A study by Michiardi and Reiva [8] bears some relation to what we have done. In their paper, the impact of three kinds of security exploits was investigated through simulations of networks using the *Dynamic Source Routing* (DSR) protocol [9]. Aggregate network throughput and delay were used to quantify how the network responds to increases in the number of malicious nodes in the system. Although the motivation there is similar to that of our work, the context is not.

An important piece of work by Wood and Stankovic [10] presents a qualitative analysis of a class of security exploits, namely *denial-of-service* attacks. Their work enumerated several of the possible exposures in the four main layers of the protocol stack. This paper made the important contribution of indicating different modalities of attacks on wireless networks, but it did not attempt to quantify the extent to which the attacks can influence the operation of the network.

In this paper, we apply simulation technology to assess the impact that different attacks can produce on two specific metrics of network performance. First, we consider the effect of an attack scenario on the network's *packet delivery ratio* (PDR), that is the ratio of the number of packets received to the number of packets sent. Second, we consider also the effect on the network's average *end-to-end delay* (E2ED) defined, for all the packets that arrive at their final destinations, as the positive difference between the time of packet reception and the time of transmission.

The attack scenarios we study in this paper do not exploit characteristics of the constructs of wireless ad hoc networking protocols (i.e., packets or radio frames) or their handshake sequences. We demonstrate that these attacks can be effective because the protocols adaptation mechanisms respond to changes in the availability of the radio links in the network. Since the adaptation mechanisms are essential to the formation and the maintenance of the ad hoc network, these attacks are particularly troubling. Although little can be done at the level of protocol construction to prevent such attacks, our results are a step toward quantifying the risks associate with the use of wireless ad hoc networks. We see this as an important contribution in understanding better the extent to which this emerging technology should be relied upon.

In the next section, we propose a model that can be used to describe a pattern in which the attack alternates *on-periods* with *off-periods*. The model we present abstracts away specific details such as the lengths of the on-periods and of the off-periods, the time when the attack starts and ends, and how multiple attackers synchronize their actions. This model was used in simulations of two types of on-off attacks, described in Sections IV and V for which we present an experimental analysis later in Section VI.

III. A General On-Off Attack Model

We propose a simple but expressive model of attack to describe a variety of malicious actions against wireless ad hoc network nodes. Our goal in proposing this model is not to define an analytical framework that can be used to derive expressions to quantify the effects of the attack on a network's performance metrics. Rather, our goal is to provide a clear characterization that can be used to guide the development of a *simulation model*.

The model we present here is applies to the types of attack which we can say are either in an *on-state*, when the associated action is effectively happening, or in an *off-state*, when the associated action is not happening and the network is free to operate normally. We expect that the kind of attacks that can be modeled in this way are mostly DoS and RoQ attacks.

The reason for proposing attacks which alternates on-states and off-states is as follows. An attack that stays permanently in its on-state may seem attractive at first because it may produce maximal, continuous disruption on the target network, what is attractive from the perspective of the attacker. Such an attack model has its downsides, however. First, it may expose the attack and the attacker. Second, in the event that the attack consumes battery power to achieve its goals, the permanent on-state will deplete the battery at a steady rate limiting the duration of the attack. Take for instance a *jamming attack*, which we discuss only for the sake of argument and which lies beyond the scope of the experimental analysis in this paper. Jamming is a powerful disruptive technique that can be used to break radio links in a wireless network, but one which requires an electronic device that emits a strong radio signal. If a jammer is continually on, it is more likely that one can succeed at using triangulation to pinpoint the location of the attacker. If the jammer were to alternate on-states and off-states, the effects observed on the network could still be significant and yet be mistakenly attributed to some kind of intermittent failure leaving the attack undetected. Furthermore, it would be harder to triangulate the source of the signal allowing the attack to extend for a longer period of time. The same applies in the case a jammer operates on battery power, for the alternation of on and off states will cause the battery to be depleted at a slower rate.

Assuming that the attacker's goals include the maximization of the quantitative impact of the network disturbance and the duration of this disturbance, the on-off process is justified. We propose a simulation model that embodies this characteristic by considering that attacks follow a cyclical process that starts at some point after the deployment of the ad hoc network. We now formally specify the details of this model.

Consider a single wireless node n in an ad hoc network. Let $t_{s,n}^{A}$ be the discrete instant in continuous time when an attack A is initiated on node n. Furthermore, let $t_{s,n}^{A}$ be described by a continuous random variable $T_{s,n}^{A}$ with arbitrary probability distribution such that $T_{s,n}^{A} \geq 0$.

We define an *attack cycle* as an on-period immediately followed by an off-period. Say that the length of the on-periods on this attack is determined by a random variable $A_n^{\text{on}} \geq 0$ and that the length of off-periods by another random variable $A_n^{\text{off}} \geq 0$. The length of an attack cycle is therefore determined by a random variable L defined as:

$$L = A_n^{\text{on}} + A_n^{\text{off}}.$$

We say that the instant of time when attack A on node n ends, denoted by $t_{e,n}^A$, is described by a random variable $T_{e,n}^A$ such that:

$$T_{e,n}^A = T_{s,n}^A + (K * L),$$

where $K \geq 0$ is a discrete random variable representing the number of attack cycles carried out. For the sake of simplicity, we assume that, for attack A and node n, the length of all on-periods are independent and identically distributed, as are also the lengths of all off-periods.

The characterization of attacks described above considers that node n is the target of attack A. While in the computational implementation of this model, it may be more natural to consider n as the node that suffers the effect of the attack, in others, it may be more natural to model n as the attacker itself. It must be pointed out that the model we are proposing works equally well for both alternatives. For the remainder of this discussion, however, we will consider that n is the attacker node, understanding that there is no loss of generality.

In order to complete the description of the attack model, one must specify the number of nodes that carry out an attack. If this number is greater than one, then the model must also specify if and how the attackers coordinate their actions. We address each of these points in their turn as follows.

For the entire model, we first determine a probability value p, which indicates the likelihood that a given node is an attacker. Next, to each wireless node we associate an independent *Bernoulli(p)* random variable. (Note that by definition all these random variables are identically distributed.) The number of attackers in a given instance of this simulation model is then determined by the summation of the values of these *Bernoulli*s.

Two special cases for this model are when there are zero attackers and when there is a single attacker. The case of zero attackers is trivial and the network operates normally, that is, the values of metrics that characterize its performance represent a baseline for comparison with scenarios in which there are one or more attackers. When the network has a single attacker, it operates independently of any other malicious node, there is no coordination or synchronization of activities. Clearly, it becomes more interesting, from the perspective of the analysis of different attack possibilities, when the number of attackers is two or greater. Their positions relative to each other and the coordination of malicious activities can be arranged in a number of different ways to produce maximal negative impact on the network and bear careful investigation.

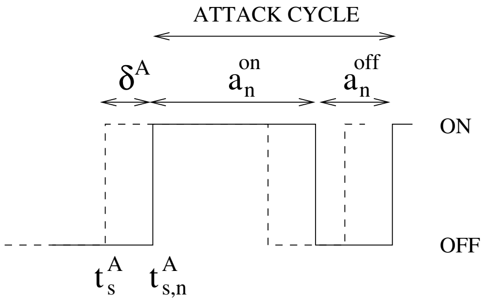

Fig. 1. Representation of an *attack cycle*.

Once the attacker nodes in a model are determined, it must be specified if and how they interact. Although the implementation of a simulation model of the attackers may resort to message passing to coordinate and even to synchronize their operations, we model this behavior using an abstraction. This abstraction allows one to do away with the details of the synchronization algorithm in the simulation model by adjusting the start time of the attack in each individual attacker. Let t_s^A be the time when attack A *might start* on the entire network. Then, we defined $t_{s,n}^A$, the time when a particular node n starts its attack is a sample from a random variable $T_{s,n}^A$ defined as:

$$T_{s,n}^A \;=\; T_s^A + \Delta^A$$

where Δ^A is a continuous random variable which defines the *synchronization window* or the *attack jitter* for A. When each attacker node uses this method to determine the starting time of its actions in attack A, the resulting effect is that all nodes participating in A will be constrained to start within the window of time $[T_s^A, T_s^A + \Delta^A]$. The attack jitter can be used to model a wide range of circumstances. When $\delta^A = 0$, the attackers are perfectly synchronized and when $\delta^A = \infty$, the attackers are completely unsynchronized (intermediate values will determine tighter or looser synchronization). The representation of an attack cycle is given in Fig. 1, where all the values indicated are samples of the corresponding random variables.

With these specifications, the parameters that define a specific attack model A for one node n are:

- The probability distribution of $T_{s,n}^A$, the time when attack A on node n starts.
- The probability distribution of A_n^{on}, the length of on-periods.
- The probability distribution of A_n^{off}, the length of off-periods.
- The probability distribution of K, the number of attack cycles.
- The probability p that some node n is an attacker (or, alternatively, the attacked node).
- The probability distribution of the attack jitter. Δ^A.

Since this framework doesn't specify the nature of the probability distributions of the various random variables, it becomes clear that one can adapt it to a wide variety of scenarios. It is clear that with these six different random variables and all the possible different combinations of probability distributions, one must be careful to construct valid and interesting attack scenarios.

We demonstrate in the remainder of this paper, that this model can be used to characterize a number of different attack scenarios. The attack models we consider relate more directly to vulnerabilities that may not be possible to counter with algorithmic constructs at the protocol level. Our main interest is in active attacks that can cause the network to either become unavailable to a subset of its nodes or to experience performance degradation. As we discuss ahead in Sections IV and V, physical actions against a network node, which can be carried out even by the most unsophisticated attacker, will cause additional traffic of control messages and affect the network's performance metrics.

IV. The Reboot Attack

The model introduced in the previous section is general enough to allow its application to several different scenarios. In this section, we illustrate this point by applying the on-off model to a scenario that we call *reboot attack*.

The idea of the reboot attack is simple. At some point in time after the network is deployed, attackers (one or more) physically capture a node and force it to power down, remain off-line for a period of time, then boot up again, and remain on-line for a period of time. We assume that the attacker has enough knowledge of the technology to identify the node's reset mechanism or perhaps to identify its power source or battery. Using only this knowledge, the actions at the attacker's disposal will be to permanently power down a node or to force it to alternate between states when it is powered up and powered down. We say that when the node is up and running, the attack is in its *off-state* and when the system is down, the attack is in its *on-state*. The pseudo-code for this attack is presented in Algorithm 1, where $U[a, b]$ represents a uniform random variate with range defined by the interval $[a, b]$.

Algorithm 1 Reboot Attack Model (pseudo-code for an attacked node n)

```
while (simulation not finished) do
  if Bernoulli(REBOOT_PROBABILITY)==1 then
    {node $n$ is under attack}
    $t_{s,n}^{\text{reboot}} \leftarrow U\left[t_s^{\text{reboot}}, t_s^{\text{reboot}} + \delta^{\text{reboot}}\right]$
    at time $t_{s,n}^{\text{reboot}}$ do:
    while (true) do
      power down and stay offline for $a_n^{\text{on}}$ sec.
      bootup and stay online for $a_n^{\text{off}}$ sec.
    end while
  end if
end while
```

The concepts behind the reboot attack and the range attack, which is described in Section V, were proposed by Chip Elliott and Stephen Boswell, from BBN Technologies, Cambridge, MA [7]. When a node is powered down it loses all forwarding information and, subsequently, when it is powered up, it will re-enter the network by first discovering any neighbors. Depending on the type of routing algorithm in its Layer 3, the node will be more or less aggressive in discovering routes and reconstructing its forwarding tables. In any case, after a node goes through one attack on-period and attempts to resume its normal operation, it is likely to communicate with other nodes to carry out the tasks of its application. In this process, since the node's memory of routes will have been erased, the Layer 3 protocol will send out control messages to discover a route on which to send application packets. The load that this control traffic will cause on the network is determined by the demands of the application layer and may cause a temporary reduction of the effective bandwidth observed in the attacked node's neighborhood.

This attack doesn't require the attackers to have any knowledge of the protocol stack running in the network nodes. One doesn't need to know which specific protocols are running or to be able to recognize the format of the data units exchanged. One doesn't have to capture cryptographic keys or to attempt to exploit authentication mechanisms. In fact, one doesn't have to know anything at all about networking or security in order to successfully carry out this attack. The precautions that would be effective against this type of attack would involve tamper-proofing the network nodes so that reset controls and battery connections wouldn't be accessible without raising some type of alarm (perhaps causing a message to be sent).

The question that we have sought to answer using simulation is under what circumstances this attack might be effective. Clearly, if only one node on the border of the network is attacked, the impact on performance metrics that determine the "health" of the network will be minimal. On the other hand, if the attacked node is a one through which many routes *must* pass, the impact of the attack will more noticeable. Assuming that attackers are poorly informed, though, it is fair to expect that they wouldn't be able to distinguish a border node from an internal node. For this reason, we assume that every node in the network is equally likely to be attacked.

Next, we present an attack model that uses the alternative perspective, that is, the attack model indicates nodes which are attackers rather than the attacked nodes.

V. The Range Attack

The propagation of the signals from one wireless network node determine what other nodes it may reach. In other words, the geographical coverage of the signal from one node depends on how far its radio transmissions can reach. There are different actions that one could take in order to interfere with the transmission range of a node. The strongest component of the signal transmitted by a node is due to the line-of-sight path. If two nodes reside on a plane and are separated by an obstacle, there is no line of sight and yet radio transmissions may go from transmitter to receiver via weaker propagation effects such as refraction and reflection.

We illustrate the concept of a *range attack* using an unusual analogy. Imagine that a wireless ad hoc network is deployed over terrain where hills and valleys abound. While some nodes may be physically close, that is, separated by small distances, they may not be able to communicate, if they happen to be separated by a small hill or a jutting rock. Imagine now, that an attacker takes possession of a node that was originally located in a valley and manages to lift it up beyond the height of surrounding obstacles. If we map the terrain to a Cartesian system of coordinates, this is equivalent to saying that the attacker will have raised the *z-coordinate* of the node without changing its *(x,y)-coordinates*. From the higher vantage point, the node is likely to gain line of sight to more nodes and therefore observe an increased "range". When the node is raised, it is possible that routes that pass through the node will be shortened by one or more hops. Routes that were shortened could become favored by the Layer 3 protocol, which would cause forwarding tables of neighboring nodes to be updated. If the attacker lowers the node at a later time, those line-of-sight links will be lost, and the Layer 3 protocol will be forced to re-adapt again. One could mount a RoQ attack on the network by simply raising and lowering nodes.

While this scenario admittedly sounds far-fetched and restricted, it is indeed a viable means to affect the performance of a running wireless network. A different way to achieve similar goals is to intermittently cover the node or its antenna with some type of electromagnetic shield. Depending on the quality of the shield, the transmission will experience more or less signal attenuation and the range of coverage will change accordingly.

Another variant of this attack is available to those with the technical knowledge to exploit what is now a common feature in modern retail IEEE 802.11 devices: transmission power control. While many IEEE 802.11 devices are designed to transmit at fixed output power, as indicated by Abdesslem et al. [11], there exist also devices which can dynamically adjust their transmission power choosing values from a discrete set of options. (Note that while nodes can be programmed to use different values of transmission power, their receiver sensitivity remains fixed.) This enhanced capability impacts wireless ad hoc networks from two different perspectives.

First, from the selfish point of view of a single node, downgrading its own transmission power allows the node to conserve battery and therefore maximize its lifetime. Since this decision may affect the node's connectivity to the rest of the network, it must be considered carefully. Second, and more importantly, when nodes can cooperatively and dynamically adjust their transmission power, the network topology can be adjusted in response to varying needs. Say for example, that a node A can reach nodes B, C, and D when transmitting at 150 mW, but can only reach node B when transmitting at 15 mW as indicated in Fig. 2. Say, also, that B can reach D when transmitting at 15 mW. If one wants A to communicate with D, while the higher transmission power setting allows the destination to be reached in one hop, the lower requires packets to be forwarded along two hops, from A to B and from B to D. Dynamic transmission power control allows the network to find its best compromise between connectivity and battery lifetime.

Although adjustable transmission power in IEEE 802.11 devices can serve a number of good purposes, it can be useful in staging a RoQ attack. One can exploit the adaptive nature of Layer 3 routing algorithms by reprogramming network nodes to alternate different transmission power settings. Algorithm 2 shows the pseudo-code for this attack. The underlying assumption is that during the attack on-period, the node is programmed to *reduce* its transmission power as opposed to the scenario originally described above. Nonetheless, a periodic change in network ensues and the topology change would cause the routing algorithm to send out route update messages that would compete for bandwidth with application data messages.

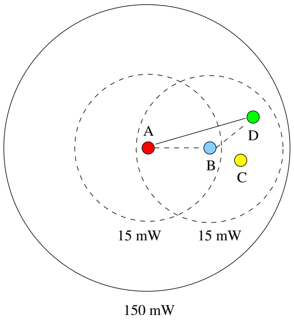

Fig. 2. Variable transmission power in IEEE 802.11 devices.

Algorithm 2 RANGE ATTACK MODEL (pseudo-code for an attacker node n)

```
while (simulation not finished) do
  if Bernoulli(ATTACK_PROBABILITY)==1 then
    {node $n$ becomes an attacker}
    $t_{s,n}^{\text{range}} \leftarrow U(t_s^{\text{range}}, t_s^{\text{range}} + \delta^{\text{range}})$
    at time $t_{s,n}^{\text{range}}$ do:
    while (true) do
      decrease TX range for $a_n^{\text{on}}$ sec.
      restore default TX range for $a_n^{\text{off}}$
      sec.
    end while
  end if
end while
```

VI. EXPERIMENTAL ANALYSIS

A. Description of the Simulation Model

Our experimental exploration in attacks on wireless networks aims for a good measure of realism. In order maximize the relevance of our simulation study, we have chosen to assign each network node the model of a complete protocol stack, which is implemented in the SWAN simulator [12]. Communication between network nodes is achieved via the RF Propagation Model, which in our simulations is represented by the Two-Ray Ground Reflection Model. Layers 1 and 2, roughly speaking PHY and MAC, conform to the specifications of the IEEE 802.11b standard and use 11 Mb/s. Unless otherwise stated, in our simulations we have considered that the default transmitter radio range is 250 m. Although our experience has indicated that the use of ARP in the protocol stack model produces small effects on the simulation, we chose to include it for the sake of completeness. For this same reason, we have included a model of the IP protocol, which implements addressing and forwarding for Layer 3. Ad hoc routing, in these experiments, is performed by a model of the AODV protocol (draft 10). By default, our AODV model uses no local repair and no HELLO messages; it relies on MAC layer transmission acknowledgments, uses expanding ring search, uses active route timeout of 10 s, and retries route requests at most twice.

The application that drives the network is modeled by a constant bit rate (CBR) source that operates at 3072 bytes per second. Although this rate is admittedly low for most modern WLAN types of applications, it is more typical in sensor networks scenarios.

Application traffic is generated in *sessions* of random length. A session is defined as a continuous stream of application packets generated for one single destination in the network. (The length of a packet is fixed throughout a simulation experiment and we investigated the effects of using packets with 512 or 2048 bytes.) The destination for one session is chosen uniformly at random among all other nodes in the network and determined at the start of the session. The traffic bit rate is used to determine the inter-arrival time of application packets. The time between two consecutive sessions is equal to the inter-arrival time between two application packets for the specified bit rate. The duration of a session is deterministic and fixed for the entire simulation experiment; we investigated the effects of session lengths of 60 seconds and 120 seconds. All nodes act both as transmitters and as receivers, so there is no concept of source and sink nodes. Traffic starts to be generated by the application layer in each node at an instant of time chosen independently and uniformly at random between 0 and 10 seconds to avoid synchronization artifacts.

In this study, we have considered only networks in which the position of nodes is stationary and deliberately chosen by human operators. The spatial distribution of nodes is a uniformly spaced mesh of 36 nodes (6 rows and 6 columns), where the distance between neighboring nodes both in the x and y axes is 150 m. The dimensions of the simulated space is 900 m by 900 m. Simulation statistics only start to be collected after the network model has been sufficiently "warmed up", after 100 seconds.

We used the following parameters for the attack models:

- The attack start time $T^{A}_{s,n}$ is deterministic and equal to 100 seconds.
- The lengths of the on-periods A^{on}_n and of the off-periods A^{off}_n of the attacks are deterministic. We experimented with four combinations of on-off values keeping the length of the attack cycle constant and shorter than the length of a session: 1/54, 5/50, 25/30, 50/5.
- The number of attack cycles K is chosen to cover the remainder of the simulation period after the attack has started.
- The probability of attack p was chosen from a set of six values: 0, 0.1, 0.2, 0.3, 0.4, and 0.5.
- The attack jitter Δ^A is defined by a uniform random variable $U[a, b]$. We experimented with perfectly synchronized attackers (jitter=0), with $U[0, 10]$ (jitter=10), and with $U[0, 100]$ (jitter=100).

Finally, in order to obtain reasonable 95% confidence intervals for the statistics estimated in the simulations, we executed 20 runs per experiment. The following network performance metrics were estimated through our statistics (their scope encompasses all the nodes in the network model):

- *Packet delivery ratio* (PDR) – the fraction of sent packets that arrives at the destination.
- *Average end-to-end delay* (E2ED) – the time it takes to move a packet from sender to receiver averaged over all the packets that arrive at their final destination.
- *AODV control packets* (CTR) – the total number of packets sent as a result of the activities performed by the routing protocol. This number includes the original route requests, route replies, routing error messages and also relayed copies of these, but no AODV HELLO messages.

B. Experiments with the Reboot Attack

One of the first points we observed with this attack is that, for a fixed packet length, the estimated PDR changes very little as we set the attack jitter to 0, 10, or 100 seconds. Fig. 3 shows how PDR varies with the probability of attack p when the on-off attack scenario is 1/54. When packet length is 512 bytes, there is no significant statistical difference between the PDR curves for the three values of attack jitter. When packet length is 2048 bytes, the point estimates for PDR values tend to be higher and lie at the upper limits of the confidence intervals for packet length of 512 bytes. This observation is counter-intuitive and needs to be investigated further in the future – we note that this behavior is repeated across all the on-off attack scenarios with which experimented. Although not indicated in this figure, we observed also that PDR curves seem to be nearly insensitive to the length of session.

Fig. 4 shows the behavior of E2ED with increasing values of the probability of attack p. Although we focus our attention in this paper on the 1/54 attack scenario due to a space restriction, we must point out that the results for E2ED are somewhat similar across two other scenarios, namely 5/50 and 25/30, and markedly different in the 50/5 scenario. As p increases, in all other scenarios, the E2ED tends to increase, while the confidence intervals for each one of the three attack jitter curves substantially overlap for each value of packet length. In the 50/5 scenario, though, the E2ED substantially decreases for values of $p > 0.3$. When attacked nodes spend the majority of time powered down, they are most often unavailable to forward packets and generate little network traffic. The immediate consequence is that in this artificially constructed topology node density is high enough for other nodes to pick up the role of forwarding. Since the load offered to the network is smaller, however, packets tend to experience less delay.

Fig. 5 shows how the number of AODV control packets changes for different scenarios of attack jitter. For each fixed value of packet length, the number of AODV control packets sent in the whole simulation is statistically the same whether the attack jitter is $U[0, 10]$ or $U[0, 100]$. When the attacks are perfectly synchronized, however, AODV reaction tends to inject fewer control packets. This observation is justified by the fact that when the down times the attacked nodes are staggered, the nodes the AODV protocol in the up and running nodes will have the opportunity to react to broken routes. When the down times of the attacked nodes are perfectly synchronized, there will be fewer running nodes to react to link breakage and consequently fewer AODV control packets are sent.

It is interesting to notice in Fig. 5 that the six curves for the number of AODV control packets form two distinct

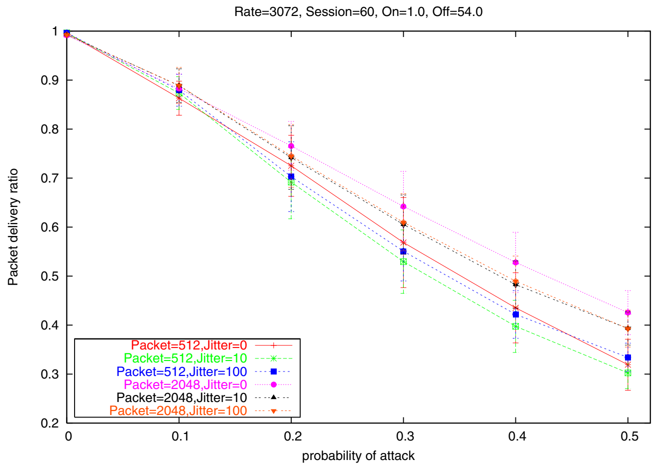

Fig. 3. Effect of reboot attack jitter on PDR.

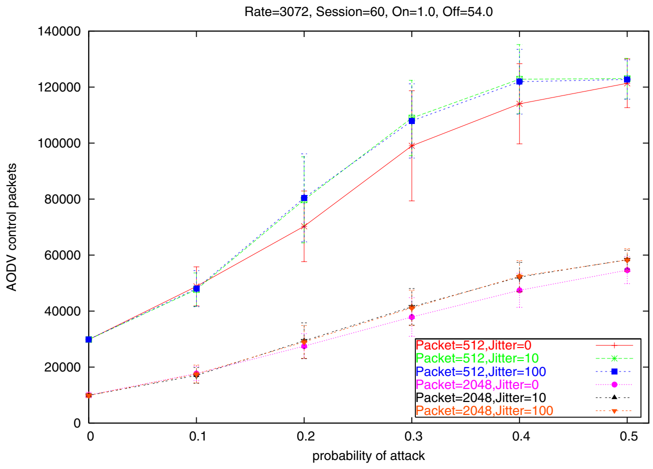

Fig. 5. Effect of reboot attack jitter on AODV control packets.

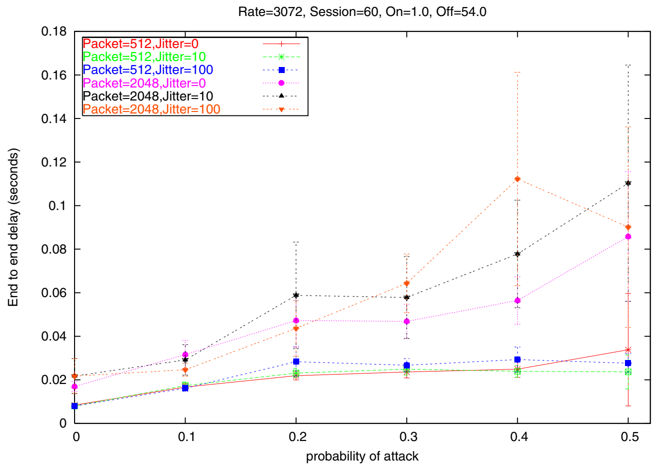

Fig. 4. Effect of reboot attack jitter on E2ED.

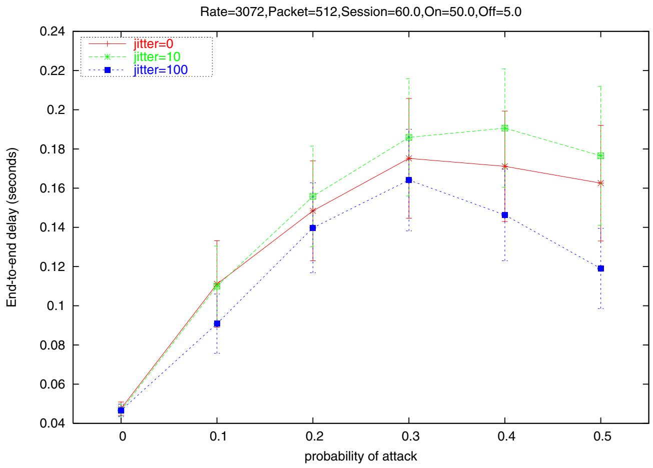

Fig. 6. E2ED with AODV HELLO messages.

groups according to packet lengths of 512 and 2048 bytes. This effect can be explained by the nature of the application that generates network traffic. With constant bit rate traffic sources that do not request the retransmission packets that do not arrive at the destination, some packets are irretrievably lost due to route breakage. Once a packet is not delivered, the time the application will push out the next packet is determined by its bit rate. For a constant bit rate, the time between the transmissions of two successive packets is proportional to the length of the packet. With smaller packet length, the application tends to cause more frequent requests for route construction or repair, what creates additional AODV control traffic relatively to the scenario with larger packets.

We repeated all the experiments with reboot attacks with AODV configured to use local repair, route reply acknowledgments, and HELLO messages. We observed that in this set of experiments, that PDR across all attack scenarios, show no significant statistical difference for varying attack jitter values. E2ED, on the other hand, behaved somewhat differently. In the 1/54 scenario with session length of 60 seconds, all curves for varying jitter are statistically close as we vary the probability of attack as in the experiments without HELLO messages. In the 50/5 scenario, the difference in E2ED behavior for varying values of attack jitter is much more pronounced, particularly when p is increased beyond 0.3, as shown in Fig. 6. This is another effect that deserves further investigation.

Figs. 7 and 8 show how the use of HELLO messages affects E2ED in the case when attackers are perfectly synchronized. In the 1/54 scenario, shown in Fig. 7, the number of AODV control packets increases with p. In this case, node outages are short and when routes going through rebooted nodes may be re-established fast enough at the cost of increased AODV control traffic. In the 50/5 scenario, on the other hand, node outages are much longer; for a much larger fraction of the total simulation time, the attacked nodes stay powered down and do not participate in route computation and contribute with fewer control packets as p increases.

C. Experiments with the Range Attack

In the simulation model for the range attack, the attacked node periodically changes its transmission power. The power values alternate between two different setting so that the transmitted signal may reach distances of 160m and 250m. Since the spatial arrangement of nodes is a regular grid or

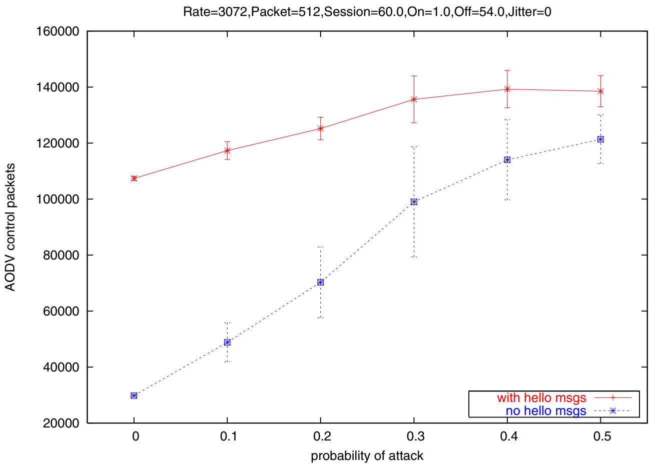

Fig. 7. AODV control messages for 1/54 reboot scenario.

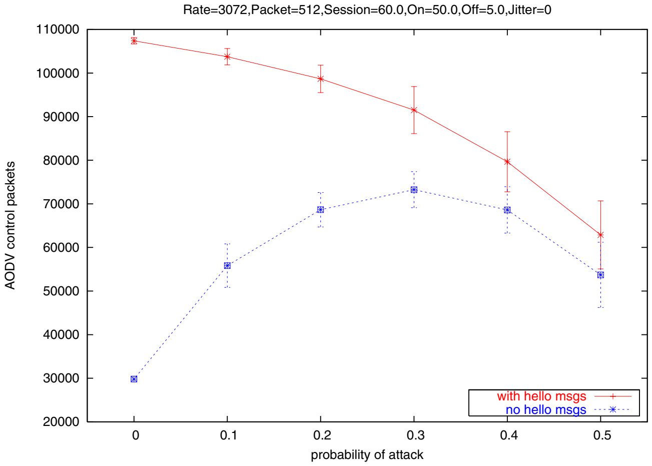

Fig. 8. AODV control messages for 50/5 reboot scenario.

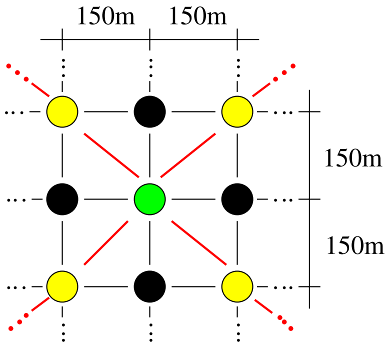

Fig. 9. Detail of the network topology used in simulation experiments.

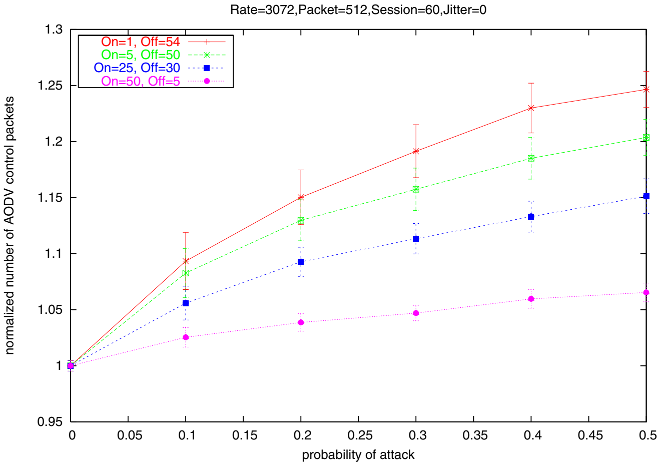

Fig. 10. The impact of the range attack on AODV control packets.

mesh, each range setting directly determines the number of neighbors that each node observes. As illustrated in Fig. 9, when the transmission range is 160m, the center node can only reach the nodes in directly above, below, to the right, and to the left (links in black). The nodes on the diagonals are positioned roughly 212m away and therefore out of range from the center node. When the transmission range is 250m, on the other hand, all the eight nodes have direct radio links to the node in the center (via the links in red).

The simulation experiments with this attack indicate that, similarly to the reboot attack model, the effects of the range attack on wireless ad hoc network are significant. Fig. 10 supports this conclusion by showing the relative increase in the number of AODV control packets for varying on-off attack scenarios in the case of synchronized attackers. We see from this plot, that a longer attack on-period produces smaller increases in AODV control traffic – this is reasonable in the light of the fact that when the attack is on, the attacked nodes have reduced transmission ranges and therefore reduced number of links. As the attack on-period decreases, the disruption on the network is long enough to trigger route updates. Since the attacked nodes' normal connectivity is restored soon, the reformed links cause even more routing traffic. The maximum effect of the attack on this metric reaches nearly 25% and corresponds to the 1/54 scenario.

Fig. 11 shows the effect of a 1/54 range attack on E2ED. We observe that in this case, perfect attacker synchronization produces only a modest increase in this performance metric. The effect of the attack is more pronounced with looser synchronization and can reach roughly up to 20% increase in E2ED.

Finally, Fig. 12, which shows the effect of the same 1/54 range attack on PDR, indicates also that looser attacker synchronization can produce more pronounced effects on this metric. In this case, however, the maximum degradation is only about 2.5%, which corresponds to a very modest impact.

VII. Conclusions

In this paper, we formalized a general attack model on wireless networks which involves the combination of cyclical on-off processes. The attacks we considered are such that they experience an on-period, in which they actively seek to disrupt the operation of the running network, followed by an off-period, in which the network experiences a transient due the adaptation mechanisms built into its protocols.

We presented a number of results of experiments with two on-off attack models. The first, which we call reboot attack, works by forcing network nodes to alternate between powered

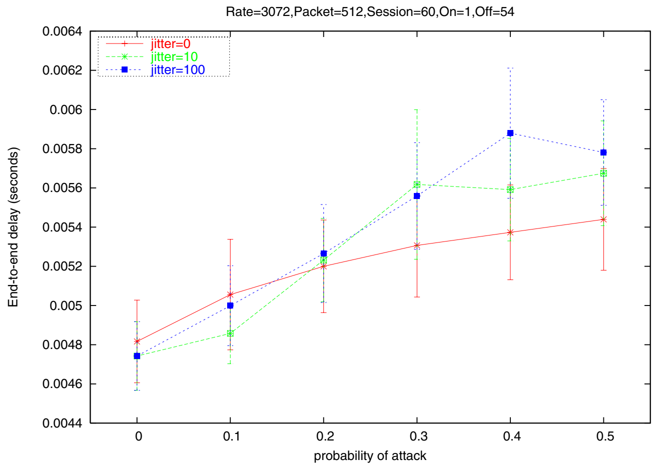

Fig. 11. The impact of the range attack on E2ED.

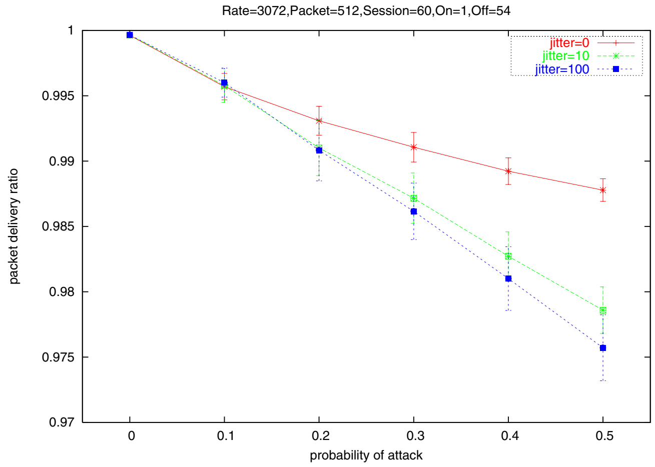

Fig. 12. The impact of the range attack on PDR.

up and powered down states. The second, which we call range attack, works by alternating the transmission range of the attacked node between two values. In both cases, as the attacks alternate between on and off periods, the connectivity of the network suffers substantial changes causing the Layer 3 protocol to seek to repair broken routs and/or to discover new ones.

The results we reported in this paper are only a small subset of the numerous simulation experiments we conducted. Since the possible combination of parameters values yielded a very large number of experiments, what we presented arose from an initial exploration of the results. Still, we were able to show that both reboot and range attacks can produce significant disruptions on the network affecting performance metrics such as end-to-end delay, packet delivery ratio, and offered number of control packets.

Future work in this topic should investigate the impact of the attack models when the lengths of the attacks' on-off periods is described by stochastic processes. When the attacks modeled depend on the direct actions of a human being, as in the case of the reboot attack, modeling the lengths of on and off periods by stochastic random variates will add an increased measure of realism: It is unreasonable to expect that a human attacker would follow precisely the same timing in every attack cycle. The stochastic element in the length of these cycles could present interesting interactions with other nodes.

The parameter space in the combination of models for the wireless network and for the attacks is very large. What we presented in this paper were the results of our initial explorations in this space and many other interesting points need to be evaluated. We will continue this work by increasing the scale of the mesh to a larger number of nodes and by studying the effect of different lengths of attack cycles on the same network protocols. The end results of these explorations in parameter space will serve to identify the worst-case scenarios that can be achieved with the attack models used and to quantify the disruption they are able to cause.

Acknowledgments

This work was supported under Award number 2000-DT-CX-K001 from the U.S. Department of Homeland Security, Science and Technology Directorate. Points of view in this document are those of the author(s) and do not necessarily represent the official position of the U.S. Department of Homeland Security or the Science and Technology Directorate.

References

[1] Y.-C. Hu and A. Perrig, "A survey of secure wireless ad hoc routing," *IEEE Security & Privacy Magazine*, vol. 2, no. 3, pp. 28–39, May-June 2004.

[2] P. Papadimitratos and Z. J. Haas, "Secure routing: Secure data transmission in mobile ad hoc networks," in *Proceedings of the 2003 ACM Workshop on Wireless Security*, September 2003.

[3] S. Yi, P. Naldurg, and R. Kravets, "Security-aware ad hoc routing for wireless networks," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, October 2001, pp. 299–302.

[4] A. Wood and J. Stankovic, "Denial of service in sensor networks," *IEEE Computer*, vol. 35, no. 10, pp. 54–62, October 2002.

[5] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad hoc on-demand distance vector (aodv) routing," http://www.ietf.org/rfc/rfc3561.txt, July 2003, [Accessed May 15, 2006].

[6] M. Guirguis, A. Bestavros, and I. Matta, "Exploiting the transients of adaptation for RoQ attacks on internet resources," in *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP 2004)*, October 2004, pp. 184–195.

[7] C. Elliott and S. Boswell, Private communication, 2002.

[8] P. Michiardi and R. Molva, "Simulation based analysis of security exposures in mobile ad hoc networks," in *Proceedings of the European Wireless 2002 Conference (EW2002)*. Florence, Italy: EUREL–Convention of National Societies of Electrical Engineers of Europe, February 2002.

[9] D. B. Johnson, D. A. Maltz, and Y.-C. H. andJorjeta G. Jetcheva, "The dynamic source routing protocol for mobile ad hoc networks (dsr)," http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt, July 2004, [Accessed May 15, 2006].

[10] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *IEEE Computer*, vol. 3, no. 10, pp. 54–62, October 2002.

[11] F. B. Abdesslem, M. D. d. A. L. Iannone, K. Kabassanov, and S. Fdida, "On the feasibility of power control in current IEEE802.11 devices," in *Proceedings of the Second IEEE Percom Workshop on Pervasive Wireless Networking*, March 2006, pp. 468 – 473.

[12] J. Liu, L. F. Perrone, D. M. Nicol, M. Liljenstam, D. Pearson, and C. Elliott, "Simulator for wireless ad hoc networks," in *Proceedings of the European Simulation Interoperability Workshop 2001 (EURO-SIW)*, June 2001.