Surviving Attacks on Disruption-Tolerant Networks without Authentication

John Burgess* George Dean Bissias† Mark Corner† Brian Neil Levine†
*BBN Technologies, Cambridge, MA, USA
†Dept. of Computer Science, Univ. of Massachusetts, Amherst, USA
jburgess@bbn.com, {gbiss, mcorner, brian}@cs.umass.edu

ABSTRACT

Disruption-Tolerant Networks (DTNs) deliver data in network environments composed of intermittently connected nodes. Just as in traditional networks, malicious nodes within a DTN may attempt to delay or destroy data in transit to its destination. Such attacks include dropping data, flooding the network with extra messages, corrupting routing tables, and counterfeiting network acknowledgments. Many existing methods for securing routing protocols require authentication supported by mechanisms such as a public key infrastructure, which is difficult to deploy and operate in a DTN, where connectivity is sporadic. Furthermore, the complexity of such mechanisms may dissuade node participation so strongly that potential attacker impacts are dwarfed by the loss of contributing participants.

In this paper, we use connectivity traces from our UMass DieselNet project and the Haggle project to quantify routing attack effectiveness on a DTN that lacks security. We introduce plausible attackers and attack modalities and provide complexity results for the strongest of attackers. We show that the same routing with packet replication used to provide robustness in the face of unpredictable mobility allows the network to gracefully survive attacks. In the case of the most effective attack, acknowledgment counterfeiting, we show a straightforward defense that uses cryptographic hashes but not a central authority. We conclude that disruption-tolerant networks are extremely robust to attack; in our trace-driven evaluations, an attacker that has compromised 30% of all nodes reduces delivery rates from 70% to 55%, and to 20% with knowledge of future events. By comparison, contemporaneously connected networks are significantly more fragile.

Categories and Subject Descriptors

C.2 [**Computer Communication Network**]: Network Protocols— *Routing Protocols*

General Terms

Security, Performance

Keywords

DTN, deployment, mobility, routing, security

1. INTRODUCTION

Disruption-tolerant networks (DTNs) provide communication in scenarios that challenge traditional mobile network solutions. DTNs use the inherent mobility of the network to deliver messages in the face of sparse deployments, highly mobile systems, and intermittent power. DTN routing differs from previous networking paradigms by assuming that connectivity will be unpredictable and poor, so information must be opportunistically routed toward the final destination.

In addition to those challenges, malicious adversaries may threaten connectivity in a DTN by inserting, flooding, corrupting, and dropping messages. In traditional, infrastructure-based networks and MANETs, security is often provided by restricting participation to a specific set of authorized nodes, enforced with cryptographic keys and identity management. In such a system, an administrator certifies all nodes in the network and participants will only route messages through other authorized nodes.

However, the choice to restrict a DTN to only authorized participants incurs an opportunity cost in the form of lost nodes that would have volunteered to participate had a simpler scheme been used. The question of whether to refuse all volunteer nodes depends on the level of threat posed by attackers and what percentage of the volunteers are honest. To demonstrate this phenomenon, we simulate the effects of adding 12 more nodes to a DTN of 18 existing authorized (and honest) nodes. The straight line in Figure 1 shows the performance of the network when only the 18 authorized nodes are available; if we increase the size of the network with 12 unauthorized but honest nodes, the average number of packets delivered per node improves by a factor of 7. As a larger proportion of the volunteers attack the network, performance degrades; however, because attacking DTNs is difficult for attackers, the network benefits from unrestricted use.

We believe that for many non-military scenarios, it is unlikely that a network will attract such a large percentage of attacking nodes. The most widely deployed peer-to-peer scenarios do not see such denial-of-service statistics, including BitTorrent, SETI@home, and Tor [10]. Therefore, in this paper we suggest that successful DTNs will encourage participation and lack authentication restrictions.

There are several other reasons to avoid authentication schemes for DTNs. Such mechanisms imply administrative registration and key distribution ahead of deployment; however, DTNs can span hundreds of miles and many administrative domains, having a common or cooperative administrative authority for all users is unwieldy. Distributed

This work was supported in part by NSF awards CNS-0133055, CNS-0519881, and CNS 0447877.

MobiHoc'07, September 9–14, 2007, Montréal, Québec, Canada.
Copyright 2007 ACM 978-1-59593-684-4/07/0009 ...$5.00.

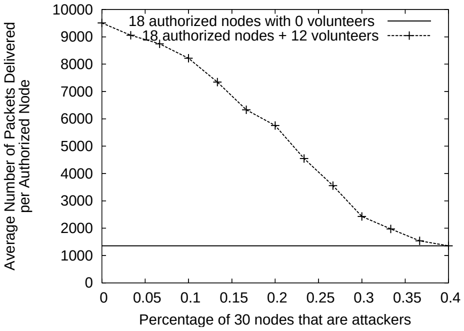

Figure 1: 12 volunteers are added to a DTN of 18 authorized nodes; the straight line represents the performance of the network when only 18 honest nodes are available. The details of this simulation are included later in the paper and correspond to the greedy case in Figure 11, where attackers use knowledge of future events to plan their attacks.

reputation schemes have been formally proven to be unworkable as well [8], and they are particularly problematic in a DTN where mobility leads to fleeting relationships with little chance for reputation building. For DTNs that do share an administrative authority, routing delays prevent querying of a public key infrastructure (PKI) supported by a central authority or distributed servers. Finally, all of these problems contribute to the difficulty of managing key revocation in a timely manner.

In this paper, we evaluate the success of attacks on DTN routing, finding that such networks are difficult to attack even when unauthorized, malicious nodes are allowed to participate. In particular, the routing protocols have been designed with an expectation that nodes are often unavailable — attacks are similar to network failures and the DTN implicitly routes around them. Moreover, the disconnected nature of the networks limits the effectiveness of attackers attempting flooding or dropping. The combination of these factors renders DTNs much less fragile than MANETs. This is not to say that a DTN's absolute performance is better than a MANET's — rather that a DTN that is without access restrictions for unauthorized nodes degrades more gracefully under attack.

One of the major themes in this paper is the two-fold benefit of epidemic-style packet dissemination in DTN routing: improved packet delivery rates and greater attack tolerance. We refer to any protocol that allows multiple copies of a given packet as *replicative*. In contrast, protocols that allow at most a single copy of each packet in the network at a time are called *forwarding*. Burgess et al. [5] showed that using the MaxProp protocol, replicative routing can perform well in terms of delivery rates. We show that MaxProp can also offer significant attack tolerance. Moreover, replicative routing is shown to be crucial to achieving this tolerance.

Contributions. We describe numerous attacks that are possible against DTN routing protocols, including dropping packets, flooding nodes with useless data, falsifying routing tables, and counterfeiting message acknowledgments. We quantitatively demonstrate the impact of attacks and countermeasures using traces of movement and transfers from a deployed vehicle-based DTN named UMass DieselNet [5] and using traces recorded by the Haggle project of a Bluetooth-based pedestrian DTN. Simulations run on these traces show evidence that replicative protocols like MaxProp [5] are more robust to attack than forwarding protocols.

We evaluate two types of attackers, *weak* and *strong*, that represent endpoints of a spectrum of possible adversaries. A weak attacker lacks global knowledge of DTN topology and transfer opportunities and is forced to choose participants at random to attack. Such a strategy is not efficient at attacking DTNs: a network where 10% of participants are attackers still achieves over 90% of its unassailed delivery rate, and it achieves over 70% of its rate when 30% are attackers.

On the other hand, we provide the strong attacker with knowledge of future events. Even with such knowledge, we prove that identifying the *most* damaging attack on a DTN is an NP-hard problem given a broad class of metrics. This result limits both a potential attacker and our own analysis. Accordingly, we adopt an attack heuristic that seeks to most lower the number of temporally connected pairs of nodes in a DTN. The strong attacker has more success: the network achieves 70% of its delivery rate when 10% of the network are attackers and only 50% of its delivery rate when 30% are attackers.

While our simulation results are limited to the protocols that we evaluated, we believe many of our conclusions hold in general for the numerous DTN routing protocols that have been proposed. Moreover, our proofs of complexity and description of possible attacks are also widely applicable.

2. BACKGROUND

Our work is related to a broad set of research on securing MANETs that restrict participation to authorized peers. These protocols make use of a PKI [24], off-line certificates [27], *a priori* key distribution [15], or a reputation system [4]. All these mechanisms are difficult to deploy in a DTN where peers expect to be frequently isolated and mobile and have difficulty depending on a central authority. The geographic span of DTNs suggests challenges in deploying a central authority when the mobile peers and volunteers do not all originate from one central location and administration.

For example, DieselNet spans 150 square miles and a large number of towns, including five colleges. The local students are not allowed Internet access on the other campuses, which is a simpler service case than a DTN despite other, decades-old formal agreements and projects between the schools. Adding volunteers from this population spread across the large geography to the DTN, including collecting authentic and verifiable credentials and protecting against Sybil attacks [11], is a serious challenge. Moreover, the delays in DTNs also exacerbate key revocation problems.

Finally, we note that symmetric reputation schemes, where peers exchange observations, have been proven to fail under simple attacks for MANETs and wired networks [8]. Asymmetric reputations schemes, where nodes form valuations based only on direct observation, are not likely to converge quickly enough in a DTN [26].

For DTNs, a distributed approach to PKI mechanisms has been proposed by Capkun et al. that leverages mobility properties to facilitate the secure delivery of data in a self-organized network [7]. However, that scheme requires a wire or infra-red wireless link between peers and user participation (to manually authorize the other peer). We are interested in DTNs where participants do not stop moving to initiate transfers, which includes vehicular and pedestrian scenarios.

The work most similar to ours is by Seth and Keshav [29]. They propose the use of hierarchical Identity Based Cryptography (IBC) [3] for end-to-end security. We suggest that the scheme could be deployed by a subset of peers in the DTN that wish to have end-to-end authentication at the application level, but it is not appropriate for secure DTN routing. As argued in the introduction, any such system requires that all nodes participate in the identity management system, potentially reducing the performance of the network. The scheme does not prevent nor detect authorized nodes acting maliciously (i.e., insider attacks), including dropping all packets, flooding of data, inversion of routing tables, and delivery acknowledgment counterfeiting. Additionally, if the root Private Key Generator (PKG) is compromised, then all keys in the system are compromised, which creates a single point of failure (they assume all PKGs are trusted and cannot be compromised). New users cannot register from within the DTN without manual verification from a kiosk operator who makes use of tamper resistant hardware. Finally, we note that they offer a proposal and not a quantitative evaluation of their approach.

Our focus is on DTN protocols based on propagation of data by means of packet replication to improve delivery rates. Many protocols are based on methods for prioritizing deliverable packets [9, 6, 13, 22, 28]. More recent routing methods have leveraged locality, mobility, delivery reports, and connectivity patterns to efficiently deliver packets [5, 6, 13, 21, 28, 33, 34, 1]. Others intentionally structure the network to increase performance using ferries on dependable schedules [34], stationary throwboxes [35, 2], or robotic autonomous agents [6].

While packet replication can alleviate the decision of path choice, it also reduces goodput under ideal network conditions. As a result, many protocols strive to minimize its use, adopting a forwarding strategy instead [17]. This is most similar to routing in conventional MANET and wired networks, and ideally the most efficient approach in a DTN. Here, we show that it is also the least robust to attack.

In this paper, we make use of public traces from two experimental DTNs: the DieselNet [5] testbed and the Haggle Project [16]. There are other DTNs that are being constructed or have been operational for a period of time [25, 31, 18]; unfortunately, none make their data publicly available.

We restrict our scope to pure DTNs. It's likely that many MANET routing schemes will have a *de facto* DTN in the areas surrounding a primary, densely populated area. Therefore, our study shows how well these routing protocols will survive when all security mechanisms are unavailable. However, we don't test existing secure MANET routing protocols because none are designed for DTNs in terms of routing — they would simply fail to create routes.

3. SYSTEM MODEL

Evaluating how well DTN routing protocols weather attack requires that we make assumptions about the size, connectivity, and mobility. In this section, we detail these assumptions and describe the routing protocols that we evaluate.

3.1 Mobility Model

In this paper, we forgo artificial models such as the random-waypoint model [20], and we base our findings on realistic mobility and connectivity patterns drawn from functioning DTNs. Both DieselNet [5] and Haggle project [16] networks have a high degree of mobility, and nodes tend to meet a large number of unique peers. We use 60 days of traces available from the DieselNet and the 3 days of traces available from the Haggle project.

DieselNet is comprised of roughly 30 buses (with the specific number varying according to bus schedule) outfitted with wireless transmitters and receivers communicating via the 802.11 protocol. Connection events occured when two busses were within range and successfully transmitted data. The 60 days of DieselNet traces describe connection events and their throughput[1].

The Haggle traces were drawn from a human mobility experiment at Infocom 2005, using 41 volunteers carrying iMotes that connected to one another, as well as connecting to Bluetooth-capable devices in the environment. To allow better comparison of Haggle data to DieselNet data, we removed connection events from the Haggle data that lasted less than one second or involved the singular appearance of a node since meaningful data transfer is likely to require setup time and nodes incapable of routing data may be ignored. After these transformations, we were left with events involving the 41 Class 1 devices in the trace and none of the Class 2 devices (which include cell phones, PDAs, etc.). Finally, we note that Haggle data, unlike DieselNet data, only reports contact times and durations without sending any data. To compare the bandwidth distribution among nodes between the two networks, we applied a constant transmission rate to Haggle data equal to median bandwidth observed in DieselNet.

There are two characteristic differences between the DieselNet and Haggle data. First, in Haggle, the median number of peers contacted by each node during the traces is 17, about 40% the network. In DieselNet, the median is 6, or only 20% of the network (see Figures 2 and 3). Further, since nodes in DieselNet seldom contact more than 10 nodes daily, few nodes directly connect to greater than one third of the network in any given day. This connectivity indicates that Haggle nodes mimic the random-waypoint model connectivity more closely than DieselNet nodes do.

Second, Figure 4 shows that 25% of node pairs in DieselNet have zero transfer bandwidth, indicating they never meet; less than 10% of node pairs never meet in Haggle. We also see from the graph, that 20% of node pairs exchange at least 60 MBs each day, while in Haggle, 20% of nodes exchange at least 80 MB each day. In the Haggle network, bandwidth distribution is more uniform, with nodes having a wider range of available bandwidth quantities.

More statistics on both DieselNet and Haggle are available from their original publications [5, 16] and elsewhere [32].

3.2 Routing Model

The routing protocol used in a DTN strongly influences the security properties of the system. We nominally identify two characteristics in routing protocols: *criterion* and *style*. The criterion refers to the process by which neighboring nodes are passed packets; specifically, we consider *metric-based* and *random* criteria. The style indicates whether the protocol is replicative or forwarding. We compare the performance when under attack of MaxProp [5] (metric-based

[1] Traces are available from `http://traces.cs.umass.edu`.

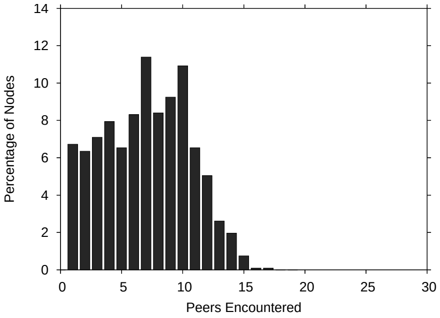

Figure 2: DieselNet: Unique Peers Connected Daily (out of 30)

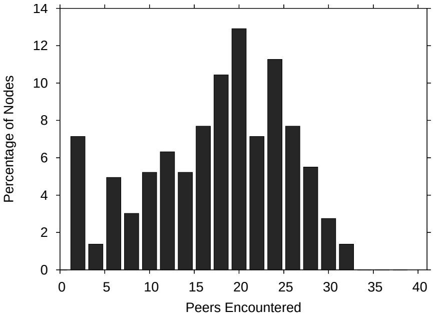

Figure 3: Haggle Project: Unique Peers Connected Daily (out of 41)

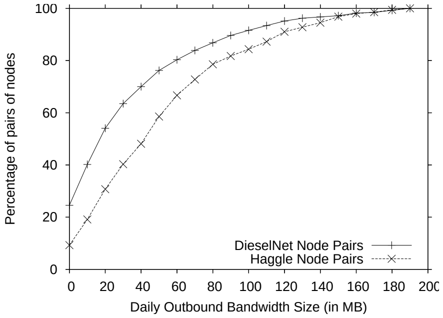

Figure 4: DieselNet: CDF of bandwidth between node pairs

Table 1: Routing Protocols

	Replicative	**Forwarding**
Metric-Based	MaxProp	MaxForw
Random	RandProp	RandForw

and replicative) to three other protocols: RandProp (random and replicative), MaxForw (metric-based and forwarding), and RandForw (random and forwarding) (see Table 1). MaxProp is a good point of departure because it offers better throughput than several other strategies: Random, FIFO, MV [6], Djikstra with an oracle of future transfer opportunities, PROPHET [23], and Spray-and-Wait [30] (as reported recently [1]). We are not aware of previous work that compares replicative and forwarding routing. As we show, replication has a number of advantages over forwarding.

MaxProp is by design a greedy protocol: it transfers as many copies of stored packets as possible in every transfer opportunity. Specific details of MaxProp can be found in the original paper; here we review its operation briefly.

MaxProp uses several mechanisms to create a ranked list that determines which packets are transmitted first during a transfer opportunity; packets at the end of the sorted list are deleted first when buffers are depleted. The primary factor that determines the ranked list is a delivery cost estimate assigned to each destination. The cost is based on the probability that the next transfer opportunity with a particular peer, estimated from observed history. These probabilities are added to form a path score; the minimum score of all possible paths via the current peer to a destination is chosen as the cost estimate. Additionally, MaxProp assigns a higher rank to new packets, and it attempts to prevent reception of the same packet twice at intermediaries. In addition, MaxProp deletes packets when acknowledgments are received, indicating a packet has been delivered. The acknowledgments are cryptographic hashes of each packet's contents. These acks, along with the routing tables that each node constructs, are propagated with replication, but are restricted to a small percentage of the bandwidth used during transfers.

4. ATTACK MODALITIES

Our goal is to determine how the network performance of a DTN degrades when no authentication protocol is used. This depends on starting with a set of assumptions about the security model and what attacks we consider. We recognize that one can construct a different set of assumptions that will cause the DTN to perform extremely poorly even with a small number of attackers. For instance, if node mobility is extremely low, and one node forms a nexus for all routing paths, the DTN will fail to deliver packets after corrupting that node. Similarly, if one attacker can corrupt all nodes by flooding an area with RF noise, the DTN will also fail. Rather, it is our intention to show that at least some DTNs have mobility patterns that perform extremely well under attack, and before applying the complexity of an authorization mechanism, one should consider whether the network really requires it.

4.1 Security Model

We have chosen to use a security model that provides a convincing case for the robustness of DTNs. This model includes several elements.

Identity: In a DTN environment without authentication, no assumptions can be made about the identities or intentions of other peers. Moreover, attackers can spoof their MAC layer addresses to appear to be any node at any time, including the destination of packets.

Routing Security: Our model only evaluates the security of the routing itself. While routing may be accomplished without authentication, this does not obviate the need for end-to-end authentication and confidentiality mechanisms. We also ignore any attacks on the applications themselves, such as spoofing requests that cause legitimate nodes to flood other legitimate nodes with unneeded traffic.

Knowledge: We distinguish between *weak* and *strong* attackers. Nodes are chosen to be weak attackers uniformly at random to simulate an opportunistic attack in real wired or wireless networks. Such opportunities may arise due to mobility, i.e., passing an infected node, or chance weakness, seen in the propagation of botnets. In contrast, strong attackers have knowledge of the complete network topology, which is likely to be more information than any node would have in practice. These two versions of our attacker provide points of reference for what is possible for attackers in a DTN. We expound upon the analysis of weak and strong attackers below.

Mobility: An attacker can follow any mobility pattern and attack all nodes that move within wireless range, or she can remain permanently within range of one node in the network. We call the latter approach a *parasite attack*; it is the most effective use of the attacker's resources.

Attack types: There are two forms of the parasite attack: node corruption and tailgating. Under node corruption, an attacker has completely taken over a node and can command it to create and drop packets at will. In some settings, this attack can include physical destruction of the target node. The tailgating attacker is external to the uncorrupted node, but can arbitrarily give it extra packets to forward, spoof outgoing packets, or selectively interfere with the delivery of packets during connection opportunities. The tailgating attacker is as powerful as node corruption since the effect of both is identical.

Given this security model, a number of attacks are possible. To simplify analysis, we focus on a set of actions that are fundamental to any attack. These four actions are detailed below in the context of DTN transfer opportunities: exchanging packets, exchanging routing tables, and exchanging acknowledgments. When a network is restricted to authenticated, authorized participants, one would expect these attacks are avoided or at least attributable to some entity.

- Dropping all packets
- Flooding of packets
- Routing table falsification
- Counterfeit acknowledgments of delivery

Note that other attacks can be formulated as a combination of these actions. For instance, corruption can be thought of as a combination of dropping and flooding and MAC layer interference is equivalent to dropping.

These attacks are described in more detail below. First, however, we explore the capabilities and limitations of the strong attacker.

4.2 The Strong Attacker

A weak attacker must, by definition, act opportunistically. Therefore, it is realistic to model compromised nodes by random assignment. In contrast, there is no clear model of behavior for the strong attacker. This makes evaluating the worst-case effects of node removal on packet delivery rate much more difficult. In response, we focus on worst-case analysis of much simpler metrics with highly salient characteristics. A connectivity based metric is ideal because it is simple to compute and any quality-of-service metric, such as delivery rate or latency, reduces to connectivity in the limit that a network becomes disconnected. Moreover, we show that the DTN vulnerability problem is NP-hard in terms of a connectivity-based metric. Thus, quality-of-service metrics are seen to be NP-hard by transitivity.

4.2.1 Graphical Model

Since a DTN is only intermittently connected, it's not possible to represent the temporal aspects of the network as a static graph. Therefore, it is sometimes useful to *flatten* the temporal graph into a static graph.

Let $D = (N, C)$ be a list of connection events C on a set of nodes N in some DTN network. The flat graph $\mathcal{G}(D) = G(V, E)$ is formed in the following manner: each node $u \in N$ is assigned a unique vertex $\mathcal{V}(u) \in V$. For any two nodes $u, v \in N$: if $(u, v) \in C$, then $(\mathcal{V}(u), \mathcal{V}(v))$ is placed in E. It is, therefore, possible to compute $\mathcal{G}(D)$ in time linear in $|C|$.

4.2.2 Problems and Complexity

In analyzing strong attacks, we evaluate the following connectivity metric.

Definition 1. A pair of nodes (i, j) are *temporally connected* in D if it is possible to construct a path between i and j by a temporally nondecreasing sequence of connection events in C.

Definition 2. The *total reachability* of a DTN D, denoted $R(D)$, is the number of pairs of temporally connected nodes (excluding reflexive pairs) in D.

We next present a result that demonstrates the computational intractability of the graph vulnerability problem under a broad class of metrics, including total reachability. This result justifies our use of heuristic analysis in Section 5.2. Perhaps more importantly, it also suggests that the graph vulnerability problem is an inherently difficult one in most every form. We begin by recalling the Vertex Cover problem.

Vertex Cover (VC)

- Input: Graph $G = \langle V, E\rangle$ and integer $k \leq |V|$.
- Output:
  - *1* if there exists a set $S \subseteq V, |S| \leq k$, such that every edge in $e \in E$ has at least one endpoint in S.
  - *0* otherwise.

The Vertex Cover problem has been shown to be NP-complete [12]. We next introduce a bit of terminology and then a generalized graph vulnerability decision problem.

Definition 3. Let $G = \langle V, E\rangle$ be a graph and $S \subseteq V$. We denote by G^S the graph resulting from the removal of vertices S and all edges incident to vertices in S.

Definition 4. The set of all graphs is given by $\mathcal{G}$. A function on $\mathcal{G}$ is any function of the form $f : \mathcal{G} \rightarrow \mathbb{R}$. We call f *well defined* if it achieves a single global minimum when G contains no edges.

Vertex Vulnerability (VV)

- Input: Graph $G = \langle V, E\rangle$, an integer $k, k \leq |V|$, $c \in \mathbb{R}$, and $f : \mathcal{G} \rightarrow \mathbb{R}$.
- Output:
  - *1* if there exists a set $S \subseteq V, |S| \leq k$, such that $f(G^S) \leq c$.
  - *0* otherwise.

THEOREM 1. *Vertex Vulnerability is NP-hard whenever $f(G)$ is well defined and computable in time polynomial in G and k.*

PROOF. Let graph $G = \langle V, E\rangle$ and integer k constitute an instance of the Vertex Cover Problem. For a given f, construct a corresponding instance of the Vertex Vulnerability problem by leaving k and G unchanged and letting $c = f(\langle V, \emptyset\rangle)$.

Suppose that $\text{VC}(G, k) = 1$ for some set $S \subseteq V, |S| \leq k$. This implies that there are no edges in G^S. Hence, the same choice of S will render $f(G^S) = c$. Therefore, $\text{VV}(G, k, f, c) = 1$.

Conversely, suppose that $\text{VC}(G, k) = 0$. It must be the case that $\text{VV}(G, k, f, c) = 0$ as well because if it did not then there would be some set $S \subseteq V, |S| \leq k$ such that $f(G^S) \leq c$. But since f is well defined we know that G^S contains no edges. This set S could thus be used to form a vertex cover for G of size less than k. Hence, $\text{VC}(G, k) = 1$. It follows by contradiction that $\text{VV}(G, k, f, c) = 0$. □

Theorem 1 applies to any graph under a wide variety of metrics. We next apply this result to the total reachability metric.

COROLLARY 1. *Determining the set of k nodes in D whose disconnection will render the lowest $R(D)$ is NP-hard.*

PROOF. We operate on the flat graph $G(V, E) = \mathcal{G}(D)$ since any problem on G naturally reduces to a problem on D when all connection events occur simultaneously. Moreover, the reduction is a polynomial-time task according to the reasoning in Section 4.2.1.

In this context, the total reachability of G, denoted $R(G)$, is the sum of all pairs of vertices in G that are connected by a path. By Theorem 1, it will suffice to show that the total reachability metric is a well defined function computable in polynomial time. Any $S \subseteq V$ that leaves no edges in G^S will render $R(G) = 0$. On the other hand, as long as there is at least one edge in G^S, $R(G) > 0$. Therefore, total reachability is well defined. Finally, $R(G)$ can be determined in time $O(|V|^3)$ using, for example, the Floyd-Warshall algorithm. □

4.3 Investigating Attack Types

4.3.1 Dropping All Packets

The simplest attack a node can mount against delivery involves dropping all packets that it receives. This attack can be devastating to network performance when a dropping node is situated along a commonly used path. For forwarding protocols, every dropped packet is a lost packet. The best defense in a DTN against malicious packet dropping attacks is the use of multiple paths.

The algorithms we evaluate cannot detect if a node is dropping packets. This is because routing tables are based on successful transfer opportunities, not successful delivery. Authentic reporting of routes would require an authentication scheme. One could defend against this attack by detecting the presence of an outage along a path and avoiding that path in the future. Because of the attack's apparent similarity to standard network outages, proving a node malicious, rather than malfunctioning or part of a route with faulty transmission links, may be difficult or impossible. Moreover, the Sybil attack [11] will make it difficult to attach blame to particular peers.

4.3.2 Flooding

During a connection opportunity, a flooding attacker continuously sends fake data destined for any node, sourced from any node, and bearing arbitrary header flags (such as bearing a hop count of 1 in MaxProp to indicate prioritized replication). Additionally, it never forwards any packets it receives from the other node. In a wired network, flooding attacks are the basis of many denial-of-service attacks, preventing legitimate traffic from reaching a victim or overwhelming the victim with false requests. Often, only a small number of nodes, each sending as much traffic as possible, is enough to disable a victim.

However, DTN flooding is much less effective because a direct route to a destination is not always available. A wired attacker can continuously send attack data, introducing traffic equivalent to many times the traffic load of a normal node. For example, if a normal node transmits only 10GB of data out of its 100 Mbps link over the course of a day, it is averaging about 116 Kbps. If this node were to begin flooding, it could output data equivalent to 864 normally operating nodes. In a DTN, legitimate nodes flood the network to the same extent to which any attacker could hope.

4.3.3 Routing Information Falsification

Traditional networks are often susceptible to injection of erroneous routing information. This can cause routers to delay or lose packets altogether. In the MaxProp protocol, tables of node contact frequencies are propagated in an replicative fashion from each node to all other peers. Table updates are integrated into a node's routing tables if they are dated more recently than those tables the node is currently using. Tables bearing dates in the future are discarded as erroneous

Because MaxProp and similar protocols do not employ authentication, attackers can propagate erroneous information about the routing tables of any node. Similar attacks are possible in MV [6] and PROPHET [23] routing, as well as in any protocol that accepts unauthenticated routing information from other nodes. Alternatively, an attacker could spoof another node's identity and falsely increase the victim's estimation of how often that node is met, however, this will be time consuming.

In our evaluations, we invert the MaxProp routing tables as we have found empirically this is more effective than randomly setting entries, and we expect it is more damaging than setting all entries to the same value. Normally entries express, for each destination x, an estimate of the probability $p(x)$ that the next node met will be x. Therefore, our attackers propagate each entry as $1-p(x)$ to encourage use of infrequent meetings, and discourage use of frequent ones. Attackers also invert the routing tables of other nodes when they are propagated in MaxProp.

4.3.4 Ack Counterfeiting

Acknowledgments are a very effective mechanism for packet delivery in replicative routing protocols and, as as a result, they are an effective method for sabotage. Acks are small in size and require little overhead; our previous work has shown there effectiveness even in isolation [5, 1]. In MaxProp, acknowledgments of delivery are simply the cryptographic hash of the packet. Unfortunately, the cryptographic hash does prevent an attacker from propagating a false acknowledgment, which victimizes intermediary nodes that have yet to see the original packet. Victims would not receive the in-transit packet from peers during transfer opportunities, cutting off a possibly viable path to the destination.

To defend against this attack without authentication, we leverage the fact that packets should normally propagate from nodes closest to a packet's source to nodes closest to a packet's destination. Consequently, packet acknowledgments should propagate in the reverse direction. In most cases, if a node receives a packet acknowledgment before the packet it describes, it is either a malicious acknowledgment, or the associated packet never propagated to the node in question. In either case, we can safely delete the acknowledgment and not propagate it. Only in the rare case that a node receives an acknowledgment followed by the described packet will this countermeasure be detrimental to routing performance, and only in the case that no additional acknowledgments are received for that packet by the node described. We cannot always expect that acks and data will follow the same paths in every DTN possible. However, our evaluation of this simple countermeasure shows that it reduces the attacker effectiveness in half with a negligible routing performance penalty.

5. EVALUATION

Our primary goal is to evaluate how well DTNs perform in the presence of attackers — both weak and strong — given the particulars of our security model. To that end, we ran trace-driven simulations using various attack models and routing protocols. Two critical metrics are commonly used to evaluate DTNs: the percentage of packets delivered (i.e., delivery rate) and the delivery latency [19, 30, 23]. In this paper, we focus on the delivery rate as the more important metric.

In our model, every honest node is a source, destination, and intermediary. To illustrate overall network throughput, all honest nodes generate traffic destined for other randomly chosen honest nodes. Because nodes may join or leave the network at any time, some packets may never be delivered even when attackers are not present. Nodes carry a 5 MB buffer in our experiments, and packets may be deleted before delivery when the buffer is full. In all simulations, packets are 10 KB, and each node generates 12 packets/hr. For DieselNet, each point on the graphs is the mean of 180 experiments: we treated each of the 60 days of traces as a separate trial, and ran each three times with different seeds. For Haggle, we broke the traces into ten segments with each comprising roughly seven hours of trace time so as to best match DieselNet data. Accordingly, each point on the graphs for Haggle data represents 30 experiments from 10 traces and three seeds. Each trace, DieselNet and Haggle, comprised roughly 30 nodes.

We use a moderate packet load in our evaluations. Whenever load is increased in a DTN, delay increases and delivery rate decreases, no matter whether the network in under attack. Therefore, in an effort to isolate the effects of the attacker, we hold load constant.

5.1 Weak Attacks

When simulating weak attacks, we randomly assign nodes as *honest* and *attackers*. This is preferable to adding new, mobile attackers, as we do not have a model to synthetically create movement and transfer data that fits our traces — creating such a model is beyond the scope of this paper.

5.1.1 Routing Resistance

To show the effect of routing on the network's resistance to attack, we ran the DieselNet traces with four routing protocols: MaxProp, RandProp, MaxForw, and RandForw (see Table 1). MaxProp and RandProp both use replicative routing, choosing neighbors to receive packets in metric-based and random fashion respectively. MaxForw and RandForw, on the other hand, pass the singular copy of each packet to a neighboring node according to respectively, metric-based or random convention.

We simulate five different attacks — the four attacks described in the previous section and, as point of comparison, simply removing nodes from the network. Figures 5 through 8 show the results of these five attacks when launched on each of the four routing protocols.

From these evaluations, we can first draw a number of conclusions that do not relate to the type of attack.

- Both RandProp and MaxProp protocols deliver significantly greater percentages of data over RandForw and MaxForw. This is unsurprising, as forwarding is ill-suited for a highly mobile, vehicular network that has only moderately predictable movement. However, it is not obvious since replication can generally overload a network with multiple copies of packets competing for scarce bandwidth.
- RandProp performs worse than MaxProp's path selection.

Second, we can make several conclusions that relate directly to the attack methods.

- Under all attack methods except ack flooding, replicative protocols (Figures 6 and 8) are largely resistant to attack and delivery rate mimics the node removal case. Ack flooding shows itself to be the most devastating attack method, as packets are dropped upon reaching an attacker node and a flood of acks from the attacker are sent to destroy any other copies of the packet within the network.
- MaxForw and RandForw routing show uniform drops in delivery rates across packet flooding, packet dropping, and ack flooding attack methods. All three attacks incorporate dropping packets and no additional effectiveness is seen by flooding acks (since the target packet has already been destroyed) or flooding junk data to contacted peers. Route inversion against MaxForw reflects a similar phenomenon.
- Although RandProp is always worse than MaxProp, it is more resistant to attack. That is to say, the magnitude of the slopes of attack curves tend to be smaller. Nodes use replication, and use it randomly, making the network difficult to attack, as the load is distributed evenly across paths.

In all cases we see that the network is quite robust, even subject to a large number of attackers — practically speaking, if a fifth of the nodes in the network are corrupt, then there are likely larger operational problems. For comparison, imagine if 20% of the nodes on the Internet were corrupt and flooding packets — the network would come to a standstill. In particular, Holme et al. [14] show that the size of largest connected component in the graph representing the router level view of the Internet will be reduced to less than half its original size after the removal of just 0.03% of the highest-degree nodes.

MaxProp has the best performance. The most effective attack against MaxProp is ack counterfeiting, and accordingly, we next propose defenses to that threat.

5.1.2 Ack Counterfeiting Defense

As described in the Section 4.3.4, we can defend against ack counterfeiting without authentication mechanisms. To demonstrate the defense's efficacy, we subject our strongest routing protocol, MaxProp, to the ack attack with and without the defense. Figure 9 shows the result with MaxForw without defense included for comparison.

- The results show that the defense significantly reduces the effectiveness of ack counterfeiting, without using authentication mechanisms.

There is a small performance penalty for the defense when all nodes are well behaved, otherwise the defense is effective. For instance, when 20% of the nodes are attackers, the delivery

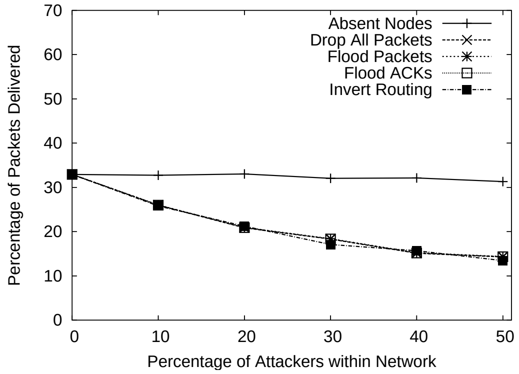

Figure 5: MaxForw

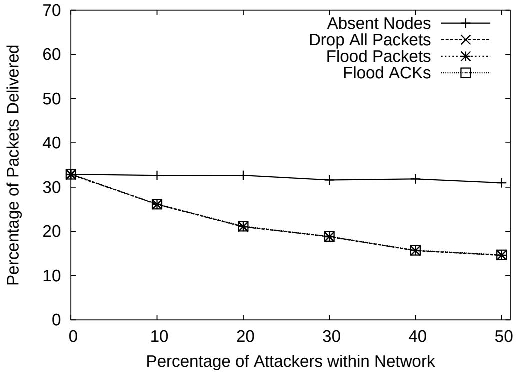

Figure 7: RandForw

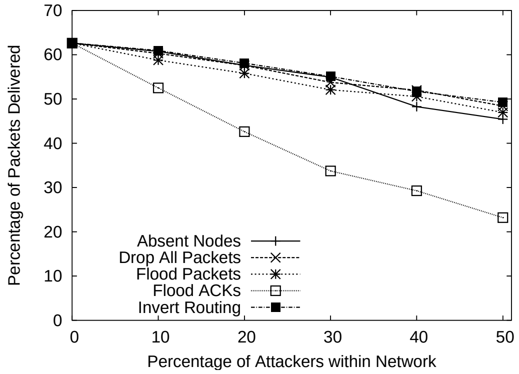

Figure 6: MaxProp

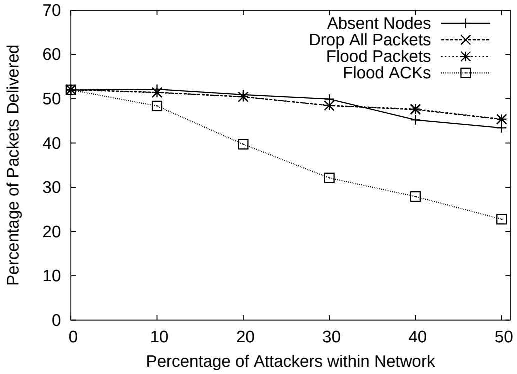

Figure 8: RandProp

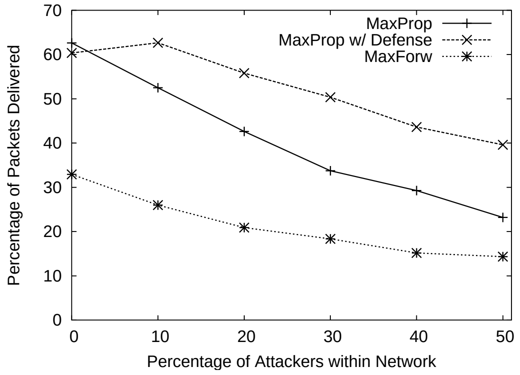

Figure 9: Metric-based routing under Ack Counterfeiting

rate in MaxProp increases from 43% to 56%. Comparing these results to the previous sections, the defense makes ack counterfeiting almost as ineffective as dropping and flooding packets.

5.2 Strong Attacks

All of the above attack scenarios have assumed that the attacker is just as likely to pick any node to compromise as any other node, which we refer to as a *weak* attacker. Attackers that can make use of topological information in selecting whom to attack are likely to be more effective at disrupting the network. Below, we evaluate a *strong* attacker that has access to the schedule of *future* connection events. Realistic adversaries will have resources that lie within the spectrum defined by the weak and strong attackers.

5.2.1 Attack Strategies

Section 4.2 demonstrated that identifying the attack with greatest affect for even the simplest metrics is an NP-hard problem. It is therefore unrealistic to identify the *most* damaging attack in terms of the routing protocols we used to evaluate the weak attacker. As a result, we focus on the *total reachability* metric, which is contained by quality-of-service metrics and for which there exists an effective heuristic. Recall that $R(D)$ denotes the number of temporally connected pairs of nodes in D. Consider the following strategies.

- **Brute:** For a given k, the brute force attack removes all possible sets of k nodes and chooses the set that most reduces $R(D)$.
- **Greedy:** The greedy attack is a recursive procedure. To greedily remove the ith node of DTN D, the attacker examines D with $i - 1$ nodes greedily removed and searches for the node that, when removed, most diminishes $R(D)$.

A brute force attack will always deliver the set of k vertices which most lower $R(D)$ but the complexity of this approach grows exponentially. Fortunately, the greedy attack appears nearly as effective. In Figures 10 and 12 we show the median impact of the Brute and Greedy attacks on seven DieselNet and Haggle traces respectively. The figures indicate that

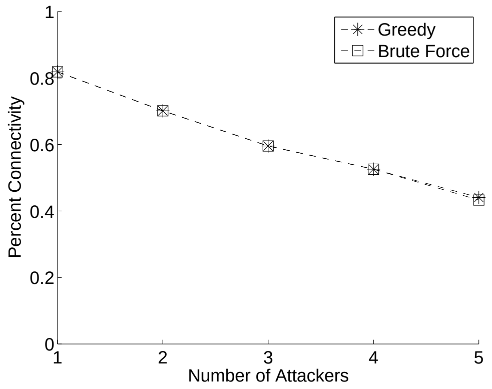

Figure 10: DieselNet - Greedy and Brute attacks

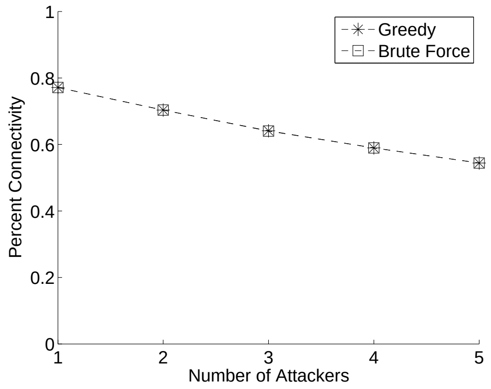

Figure 12: Haggle - Greedy and Brute attacks

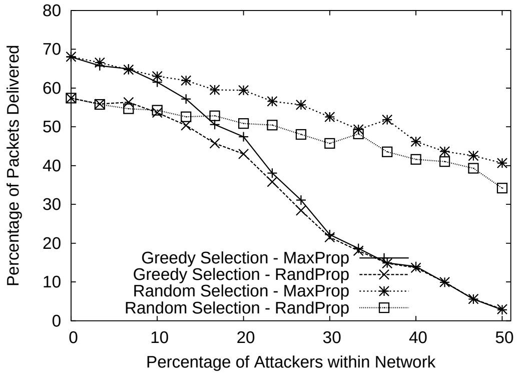

Figure 11: DieselNet - Routing Under Attack from "DropAll" Attackers

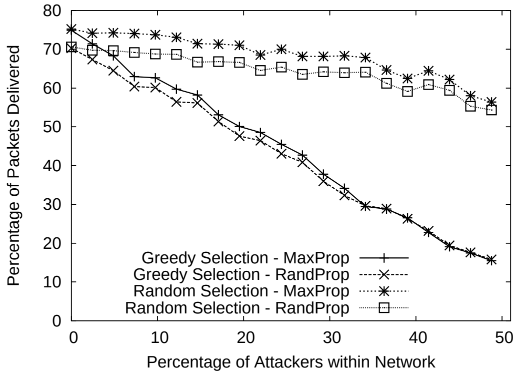

Figure 13: Haggle - Routing Under Attack from "DropAll" Attackers

the Brute and Greedy attacks nearly coincide for both types of traces up to $k = 5$. We operate under the assumption that the greedy attack will continue to model the brute force attack well for higher values of k.

5.2.2 Experimental Findings

Figure 11 shows the effect of the Drop All attack using random and greedy node selection. We can draw several conclusions.

- The results demonstrate that greedy node selection is much more effective than random selection at reducing packet delivery rates in MaxProp.
- One might expect RandProp to exhibit greater robustness than MaxProp since it evenly uses all available paths rather than one favored path. However, the results show that RandProp does not mitigate this attack either, and its performance coincides with MaxProp routing as the percentage of attackers increases. The results suggest that when the number of attackers is lower, the routing protocol used has a stronger influence on performance under attack. In contrast, when the number of attackers is higher, attack effectiveness is influenced more strongly by the connectivity patterns in the network. This suggests that the effects of greedy attack selection may be difficult to avoid, even with an enhanced routing protocol.
- With the elimination of critical routes in the network, delivery rate will tend to suffer regardless of the routing protocol.

It is possible that these results are particular to the mobility found in the DieselNet network. To demonstrate the effects on another mobility trace, we used measurements taken by the Haggle project [16]. Figure 13 shows the results of the previous experiment when conducted on the Haggle traces.

The figures show that when there are no attackers in the network, the number of independent paths in Haggle are higher than in DieselNet. We believe, from the results in Figures 2 and 3, this is due to a better "mixing" of participants at Infocom compared to a scheduled bus network. This characteristic has several effects.

- Although the trends are roughly the same in both experiments, Haggle network degrades more gradually as more nodes are attacked.

- The performance of MaxProp and RandProp are less distinguishable in the case of Haggle. The Haggle network is more random and has less structure — it is less likely that there is a single node that leads to a destination, and so RandProp will make few poor decisions.

In general, the Haggle results agree with the DieselNet results: DTNs are robust even without authentication to restrict participation to honest nodes.

6. CONCLUSION

Routing in disruption-tolerant networks is robust in the presence of many powerful attackers. The unpredictable nature of DTNs reduces the effectiveness of attacks to that of simple network failures. In this paper, we proposed a variety of attack strategies with related complexity results and introduced attack modalities with a defense for the most powerful. Using a comprehensive set of experiments, we have demonstrated that even in the worst case, of a very powerful attacker that has corrupted 20% of the nodes, a replicative DTN routing protocol still delivers 45% of all packets successfully, compared with 70% when no attackers are present. In our simulations, we found that attacks were no worse than absent nodes since attackers also have limited resources in a DTN and replication-based routing protocols already assume failure. This evaluation brings into question whether the opportunity costs of excluding unauthenticated participants is worth avoiding the negative effects of including possible attackers.

7. REFERENCES

[1] A. Balasubramanian, B. N. Levine, and A. Venkataramani. DTN Routing as a Resource Allocation Problem. In *Proc. ACM SIGCOMM*, August 2007.

[2] N. Banerjee, M. D. Corner, and B. N. Levine. An Energy-Efficient Architecture for DTN Throwboxes. In *Proc. IEEE Infocom*, May 2007.

[3] D. Boneh and M. Franklin. Identity based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.

[4] S. Buchegger and J. Le Boudec. The Effect of Rumor Spreading in Reputation Systems for Mobile Ad hoc Networks. In *Proc. WiOpt*, pages 131–140, 2003.

[5] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In *Proc. IEEE INFOCOM*, April 2006.

[6] B. Burns, O. Brock, and B. N. Levine. *MV* routing and capacity building in disruption tolerant networks. In *Proc. IEEE INFOCOM*, pages 398–408, March 2005.

[7] S. Capkun, J. Hubaux, and L. Buttyan. Mobility Helps Peer-to-Peer Security. In *IEEE Trans. on Mobile Computing*, volume 5, pages 43–51, Jan 2006.

[8] A. Cheng and E. Friedman. Sybilproof Reputation Mechanisms. In *ACM Workshop on the Economics of Peer-to-Peer Systems*, pages 128–132, August 2005.

[9] J. Davis, A. Fagg, and B. N. Levine. Wearable Computers and Packet Transport Mechanisms in Highly Partitioned Ad hoc Networks. In *Proc. IEEE Intl. Symp on Wearable Computers (ISWC)*, pages 141–148, October 2001.

[10] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The Second-Generation Onion Router. In *Proc. USENIX Security Symposium*, pages 303—320, August 2004.

[11] J. Douceur. The Sybil Attack. In *Proc. Intl Wkshp on Peer-to-Peer Systems (IPTPS)*, Mar. 2002.

[12] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman and Co., 1979.

[13] M. Grossglauser and M. Vetterli. Locating Peers With Ease: Mobility Diffusion Of Last Encounters In Ad hoc Networks. In *Proc. IEEE Infocom*, April 2003.

[14] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han. Attack Vulnerability of Complex Networks. *APS Physics Review E*, 65(5):056109.1–056109.14, May 2002.

[15] Y.-C. Hu, A. Perrig, and D. B. Johnson. Ariadne: A Secure On-Demand Routing Protocol for Ad hoc Networks. *Wireless Networks*, 11(1-2):21–38, 2005.

[16] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Pocket Switched Networks and Human Mobility in Conference Environments. In *Proc. ACM Workshop on Delay-Tolerant Networking*, pages 244–251, Aug. 2005.

[17] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *Proc. ACM SIGCOMM*, pages 145–158, August 2004.

[18] P. Juang et al. Energy-Efficient Computing for Wildlife Tracking: design tradeoffs and early experiences with ZebraNet. *SIGOPS Oper. Syst. Rev.*, 36(5):96–107, 2002.

[19] J. Kong, X. Hong, Y. Yi, J.-S. Park, J. Liu, and M. Gerla. A Secure Ad hoc Routing Approach Using Localized Self-Healing Communities. In *Proc. ACM MobiHoc*, pages 254–265, May 2005.

[20] D. Kotz, C. Newport, R. Gray, J. Liu, Y. Yuan, and C. Elliott. Experimental Evaluation of Wireless Simulation Assumptions. In *Proc. ACM/IEEE Intl Symp on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 78–82, Oct 2004.

[21] Q. Li and D. Rus. Sending Messages to Mobile Users in Disconnected Ad hoc Wireless Networks. In *Proc. MobiCom*, pages 44–55, August 2000.

[22] A. Lindgren, A. Doria, and O. Scheln. Probabilistic Routing in Intermittently Connected Networks. In *Proc. Workshop on Service Assurance with Partial and Intermittent Resources*, August 2004.

[23] A. Lindgren, A. Doria, and O. Scheln. Probabilistic Routing in Intermittently Connected Networks. In *Proc. Workshop on Service Assurance with Partial and Intermittent Resources*, August 2004.

[24] P. Papadimitratos and Z. Haas. Secure Routing for Mobile Ad Hoc Networks. In *Proc. SCS Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2002.

[25] A. Pentland, R. Fletcher, and A. Hasson. DakNet: Rethinking Connectivity in Developing Nations. *IEEE Computer*, 37(1):78–83, Jan 2004.

[26] C. Piro, C. Shields, and B. N. Levine. Detecting the Sybil Attack in Ad hoc Networks. In *Proc. IEEE/ACM Intl. Conf. on Security and Privacy in Communication Networks (SecureComm)*, pages 1–11, Aug. 2006.

[27] K. Sanzgiri, B. Dahill, D. LaFlamme, B. N. Levine, C. Shields, and E. Belding-Royer. Authenticated Routing for Ad hoc Networks. *IEEE/ACM Journal of Selected Areas in Communications: Special issue on Wireless Ad hoc Networks (JSAC)*, 23(3):598–610, March 2005.

[28] N. Sarafijanovic-Djukic and M. Grossglauser. Last Encounter Routing under Random Waypoint Mobility. In *Proc. IFIP-TC6 NETWORKING*, pages 974–988, 2004.

[29] A. Seth and S. Keshav. Practical Security for Disconnected Nodes. In *Proc. Workshop on Secure Network Protocols (NPSEC)*, Nov. 2005.

[30] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks. In *Proc. ACM Workshop on Delay-Tolerant Networking*, pages 252–259, Aug. 2005.

[31] Wizzy digital courier. http://www.wizzy.org.za.

[32] X. Zhang, J. Kurose, B. N. Levine, D. Towsley, and H. Zhang. Study of a Bus-Based Disruption Tolerant Network: Mobility Modeling and Impact on Routing. In *Proc. ACM Mobicom*, September 2007.

[33] W. Zhao and M. Ammar. Message Ferrying: Proactive Routing In Highly Partitioned Wireless Ad hoc Networks. In *Proc. IEEE Wkshp on Future Trends in Distributed Computing Systems*, May 2003.

[34] W. Zhao, M. Ammar, and E. Zegura. A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad hoc Networks. In *Proc. ACM Mobihoc*, May 2004.

[35] W. Zhao, Y. Chen, M. Ammar, M. D. Corner, B. N. Levine, and E. Zegura. Capacity Enhancement using Throwboxes in DTNs. In *Proc. IEEE Intl Conf on Mobile Ad hoc and Sensor Systems (MASS)*, pages 31–40, Oct 2006.